ECMA
EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-111

SMALL COMPUTER SYSTEM INTERFACE SCSI

December 1985

Free copies of this document are available from ECMA,
European Computer Manufacturers Association
114 Rue du Rhône – 1204 Geneva (Switzerland)

ECMA

EUROPEAN COMPUTER MANUFACTURERS ASSOCIATION

STANDARD ECMA-111

SMALL COMPUTER SYSTEM INTERFACE SCSI

December 1985

BRIEF HISTORY

ECMA set up a Working Group in 1983 to investigate the need for a standard for a Small Computer System Interface (SCSI). In 1984 Technical Committee TC30 was set up with the task of drafting such a standard. The work of TC30 was based on the work of the X3T9.2 standardization group in the USA which put at ECMA's disposal their own draft which was already considerably advanced. Further work in ECMA was conducted in close co-operation with X3T9.2.

This ECMA Standard is compatible with the ANSI standard developed by X3T9.2. However, it is more restricted in some instances, e.g. the Arbitration phase is mandatory in the ECMA Standard, so that it can be considered as a sub-set of the ANSI standard.

The Small Computer System Interface (SCSI) defines a low cost local input/output bus. The interface can be operated in synchronous mode at data rates up to 4 megabytes per second and at a distance of up to 25 m using differential drivers and receivers. For local connection within a cabinet single ended drivers and receivers allow up to 6 m of cable to be used. Data may be transferred either synchronously or asynchronously.

The SCSI provides a connection between host computers, disk drives, tape drives and printers. Communication on the bus is allowed only between two Bus Devices at any given time. A maximum of eight Bus Devices may be connected to the bus. The interface is specified to provide device independence within a class of devices. Hence different disk drives, tape drives and printers may be added to the host computer(s) without modification being required to the system hardware and software.

The Standard defines the mechanical, electrical and functional requirements of the bus as well as command sets for peripheral device types commonly used with small computers. When two devices communicate on the bus one acts as an Initiator and the other acts as a Target. A device usually acts as either an Initiator or a Target but some devices may assume either role. Although only eight devices may be attached to the bus an Initiator may address up to eight peripheral devices that are connected to a Target. A Target may disconnect from the bus at any time during a transfer and reconnect at a later time to complete the transfer in order to maximize use of the bus bandwidth. Distributed arbitration is built into the bus architecture. A priority scheme awards control to the highest priority Initiator that is contending for the bus. The time to complete arbitration (bus-contention) is independent of the number of devices contending.

Accepted as an ECMA Standard at the General Assembly of Dec. 12, 1985.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont'd)

TABLE OF CONTENTS (cont'd)

TABLE OF CONTENTS (cont'd)

SECTION I

1. SCOPE
2. GENERAL
3. CONFORMANCE

1. SCOPE

This ECMA Standard specifies the general structure and the physical and logical characteristics of an interface intended for small computer systems.

2. GENERAL

2.1 Definitions

For the purpose of this Standard the following definitions apply.

2.1.1 Bus Device

A host computer adapter, a peripheral controller or an intelligent peripheral device that can be physically connected to the SCSI bus.

2.1.2 To connect

The action by an Initiator when it selects a Target with a view to starting an operation.

2.1.3 To disconnect

The action by a Target when it releases control of the SCSI bus, allowing it thereby to go to the Bus Free phase.

2.1.4 Initiator

A Bus Device that requires an operation to be performed by another Bus Device.

2.1.5 Logical Unit

A physical or virtual device that is addressable through a Target.

2.1.6 Medium

That part of a peripheral device on which information is recorded or from which information is read, e.g. a magnetic tape, a magnetic tape cartridge, a flexible disk cartridge, a rigid disk cartridge, a rigid disk pack, an optical digital data disk, unit, paper, etc.

2.1.7 Peripheral Device

A physical device that can be physically connected to a Bus Device and containing a medium on which information can be recorded and/or from which information can be read.

2.1.8 To re-connect

The action by a Target to select an Initiator with a view to continuing an operation previously interrupted by a disconnect action by that Target.

2.1.9 Target

A Bus Device that performs an operation requested by an Initiator.

2.2 Notational Conventions

For the purpose of this Standard the following notational conventions are applied.

2.2.1 Capital Letters

All names of messages, commands, status, signals and sense keys are typed with capital letters.

The two binary digits ZERO and ONE are written with capital letters. When the number zero is meant it is written with small letters.

All acronyms are written with capital letters.

2.2.2 Capital Initial

All names of phases, bus conditions, delays, fields of the Control Descriptor Blocks as well as of all basic concepts (e.g. Initiator, Target, Pointer, etc.) are typed with a capital letter as initial.

2.2.3 Italics

All notes are typed in italics.

2.2.4 Hexadecimal Notation

Hexadecimal values are always shown between parentheses.

2.2.5 Acronyms

The acronyms for the signal names are specified in 5.1.4.

Other acronyms used are as follows.

CDB stands for : Command Descriptor Block
CRC stands for : Cyclic Redundancy Check
DB stands for : DB1 to DB8 and DP
ECC stands for : Error Correcting Character
ID stands for : Identification of a Bus Device
LSB stands for : Least Significant Byte
LUN stands for : Logical Unit Number
MSB stands for : Most Significant Byte

3. CONFORMANCE

3.1 Alternatives

This ECMA Standard contains various alternatives which are mutually exclusive within a system.

i) Single-ended or differential drivers.

ii) Termination power supplied by cable or not.

iii) Parity implemented or not.

iv) Hard or Soft Reset implemented.

v) Reservation queuing implemented or not.

By specifying which alternatives are implemented an environment is created enabling different features to be implemented. These features are divided into two nested minimum levels: a Basic and an Extended Level.

3.2 Basic Level

In order to conform to the Basic Level of this Standard the following features shall be implemented.

i) All Initiators shall implement:

- Arbitration,
- the mandatory commands of the common command set,
- the mandatory commands of the sets of commands of the device types supported,
- the COMMAND COMPLETED message.

ii) If further messages are implemented, then the COMMAND REJECTED message shall be implemented.

iii) At least GOOD STATUS and the CHECK CONDITION within the Status Bytes shall be implemented.

3.3 Extended Level

In order to conform to the Extended Level of this Standard the following features shall be implemented in addition to all the mandatory features of the Basic Level:

i) Disconnection and Reselection shall be implemented.

ii) All Initiators shall implement:

- the mandatory commands within the common command set,
- the mandatory commands of the set of commands of the device types supported,
- the DISCONNECT, MESSAGE REJECTED and IDENTIfY messages.

iii) The Attention condition shall be implemented.

3.4 Options

3.4.1 Non-mandatory Commands

This Standard specifies a number of further commands which can be implemented in either level. Some of these commands, if implemented, require the implementation of other non-mandatory commands and/or messages. In this case these additional commands and/or messages shall also be implemented.

3.4.2 Non-mandatory Messages

This Standard specifies a number of further non-mandatory messages which may be implemented at either level.

3.4.3 Other Options

Within some commands additional features called options are described which need not necessarily be implemented.

3.5 Statement of Conformance

Any statement of conformance to this Standard shall declare which of the alternatives listed in 3.1 and which of the two levels are implemented. In addition it shall indicate which of the options listed in 3.4 are implemented, if any.

SECTION II

4. PHYSICAL SPECIFICATION

4. PHYSICAL SPECIFICATION

This section contains the physical specification of the Bus Devices and covers the cables, the connectors and the bus circuits.

4.1 Cable Requirements

The Bus Devices are connected together using a common cable. Both ends of the cable shall be terminated and the terminators may be internal or external to the Bus Devices. All signals are common between all Bus Devices. It is desirable to minimize the use of cables of different impedances in the same bus in order to minimize discontinuities and signal reflections.

4.1.1 Cable for Single Ended Bus Circuits

A 50-conductor flat cable or 25-twisted-pair, round cable shall be used. The characteristic impedance should be 132 Ohm. Each conductor in the cable shall have a section of 0,8 mm^2 minimum. The maximum cable length shall be 6,0 m. Each Bus Device connection shall have a maximum stub length of 0,1 m. The connector pin assignment shall be as specified in 4.2.2.

4.1.2 Cable for Differential Bus Circuits

A 50-conductor flat cable or 25-twisted-pair, round cable shall be used. The characteristic impedance should be 122 Ohm. Each conductor in the cable shall have a section of 0,8 mm^2 minimum. The maximum cable length shall be 25 m. Each Bus Device connection may have a maximum stub length of 0,2 m. The connector pin assignment shall be as specified in 4.2.3.

4.2 Connector Requirements

The connectors are specified for both cables and Bus Devices.

4.2.1 Non-Shielded Connector

The non-shielded cable connector shall be a 50-conductor connector with two rows of female contacts on 2,54 mm centres. It is recommended to use keyed connectors.

The non-shielded Bus Device connector shall be a 50-conductor connector consisting of two rows of male pins on 2,54 mm centres.

4.2.2 Pin Assignment for Single-Ended Bus Circuits

PIN	SIGNAL	PIN	SIGNAL
1	GROUND	2	DB-1
3	GROUND	4	DB-2
5	GROUND	6	DB-3
7	GROUND	8	DB-4
9	GROUND	10	DB-5
11	GROUND	12	DB-6

PIN	SIGNAL	PIN	SIGNAL
13	GROUND	14	DB-7
15	GROUND	16	DB-8
17	GROUND	18	DB-P
19	GROUND	20	GROUND
21	GROUND	22	GROUND
23	GROUND	24	GROUND
25	OPEN	26	TERMPWR
27	GROUND	28	GROUND
29	GROUND	30	GROUND
31	GROUND	32	ATN
33	GROUND	34	GROUND
35	GROUND	36	BSY
37	GROUND	38	ACK
39	GROUND	40	RST
41	GROUND	42	MSG
43	GROUND	44	SEL
45	GROUND	46	C/D
47	GROUND	48	REQ
49	GROUND	50	I/O

4.2.3 Pin Assignment for Differential Bus Circuits

PIN	SIGNAL	PIN	SIGNAL
1	GROUND	2	GROUND
3	+DB-1	4	-DB-1
5	+DB-2	6	-DB-2
7	+DB-3	8	-DB-3
9	+DB-4	10	-DB-4
11	+DB-5	12	-DB-5
13	+DB-6	14	-DB-6
15	+DB-7	16	-DB-7
17	+DB-8	18	-DB-8
19	+DB-P	20	-DB-P
21	D-ENABLE	22	GROUND
23	GROUND	24	GROUND
25	TERMPWR	26	TERMPWR
27	GROUND	28	GROUND
29	+ATN	30	-ATN
31	GROUND	32	GROUND
33	+BSY	34	-BSY
35	+ACK	36	-ACK
37	+RST	38	-RST
39	+MSG	40	-MSG
41	+SEL	42	-SEL
43	+C/D	44	-C/D
45	+REQ	46	-REQ
47	+I/O	48	-I/O
49	GROUND	50	GROUND

4.3 Electrical Requirements

The electrical requirements given below shall be referred to the input/output connector for the Bus Device and it is assumed that the bus termination is external to the Bus Device. Single-ended bus circuits use open-collector or three-state drivers. These types of drivers shall not be mixed on a given signal line. Differential bus circuits shall use differential drivers.

	SINGLE-ENDED BUS CIRCUITS	DIFFERENTIAL BUS CIRCUITS
Output signal (low level voltage)	0,4 V max	2,0 V max
Output sink current at 0,5 V	48 mA min	-
Output sink current at 2,0 V	-	55 mA min
Output signal (high level voltage)	2,5 V min	3,0 V min
Output source current		
Output source current at 3,0 V	-	55 mA min
Input signal (low level voltage)	0,8 V max	-
Input sink current at 0,4 V	0,4 mA max	2,0 mA
Input sink current		
Input signal (high level voltage)	2,0 V min	-
Input source current	-	-
Input source current	-	2,0 mA
Input hysteresis	0,2 V min	0,035 V min

As an option, Pin 21 of the connector is reserved for an active high-enable for the differential drivers. If a single-ended receiver or terminator is inadvertently connected, this pin is grounded, disabling the drivers (see Fig. 1).

4.4 Termination

Each end of the cable shall be terminated and it is assumed that the termination is external to the Bus Devices. The Bus Devices may have the provision for an optional internal termination.

4.4.1 Termination for Single-Ended Bus Circuits

All assigned signals shall be terminated at each end of the cable wit 220 Ohn between +5 V and the Signal, and 330 Ohm between the Signal and GROUND (see Fig. 2).

4.4.2 Termination for Differential Bus Circuits

Each signal consists of two lines +Signal and -Signal. Each assigned signal shall be terminated at each end of the cable with 330 Ohm between + 5 V and -Signal, 150 Ohm between -Signal and +Signal, 330 Ohm between +Signal and GROUND (see Fig. 3).

4.4.3 Terminator Power

Bus Devices which supply terminator power via the cable shall do so through a diode or similar semi-conductor which prevents the backflow of power to the Bus Device.

	SINGLE-ENDED BUS CIRCUITS	DIFFERENTIAL BUS CIRCUITS
Supply to pins	Pin 26	Pins 25 and 26
Voltage range (TERMPWR)	4,0 V to 5,25 V	4,0 V to 5,25 V
Minimum source drive capability	800 mA	600 mA
Maximum source limitation	1000 mA	1000 mA
Maximum sink capability	1 mA	1 mA

The use of keyed connectors is recommended in Bus Devices which provide terminator power to prevent accidental grounding or misconnection to terminator power.

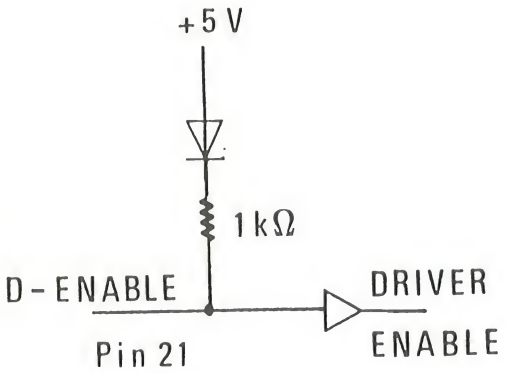


Figure 1

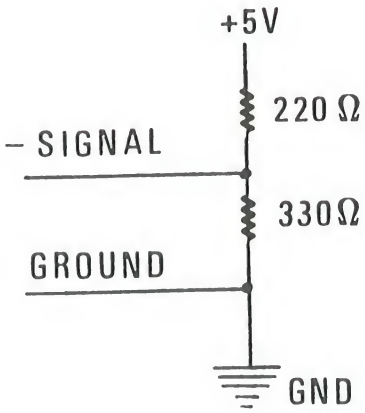


Figure 2

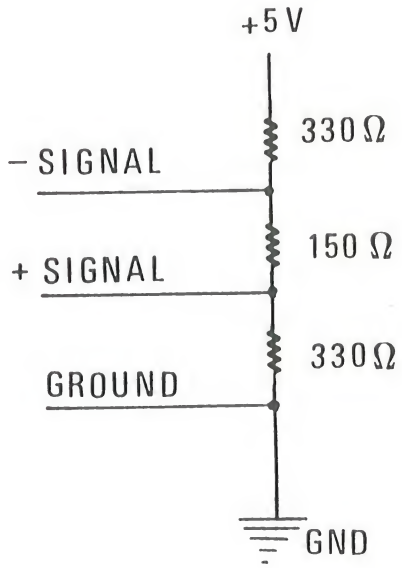


Figure 3

SECTION III

5. LOGICAL SPECIFICATION

5.1 Bus States, Phases and Signals

5.1.1 Definitions

Definitions of a signal state

The signals carried on the bus lines can only have two discrete Signal States:

- a Signal State ONE — For single-end drivers a low level voltage. For differential drivers the voltage on +Signal is greater than the voltage on -Signal.
- a Signal State ZERO — For single-end drivers a high level voltage. For differential drivers the voltage on -Signal is greater than the voltage on +Signal.

Definition of a bus phase

A Bus Phase is an ordered sequence of Signal States.

The Standard defines ten Bus Phases:

BUS FREE, ARBITRATION, SELECTION, RESELECTION, MESSAGE-IN, MESSAGE-OUT, DATA-IN, DATA-OUT, COMMAND and STATUS.

5.1.2 Bus Phase Specification

The bus has ten distinct phases:

Bus Free Phase

This phase indicates that the bus is not being used by any bus device and that the bus is available for subsequent users.

Arbitration Phase

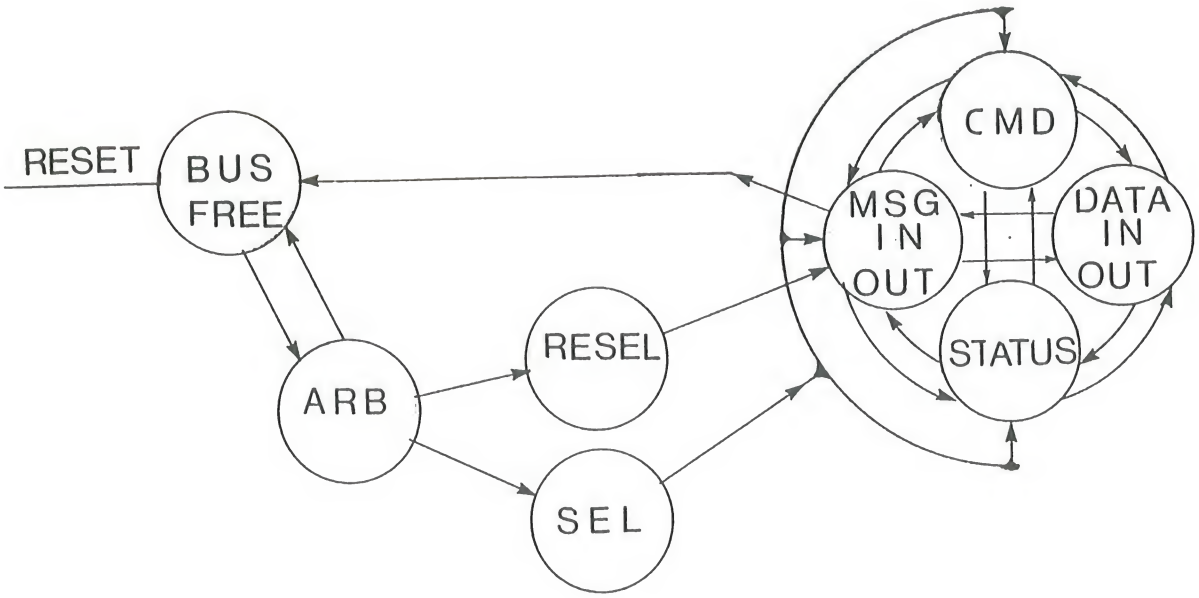
This phase is used to allocate the bus for subsequent use to one particular bus device, i.e. the bus device (the Initiator in the case of connection and the Target in the case of reconnection) with the (during this phase) highest priority bus request.

Selection Phase

This phase is used (in the case of connection) to allow the bus device to which the bus is allocated (e.g. the Initiator) to select another bus device (e.g. the Target) to establish a connection for subsequent use.

Reselection Phase

This phase is used (in the case of reconnection) to allow the Bus Device to which the bus is allocated (e.g. the Target) to reselect another Bus Device (e.g. the Initiator) to re-establish a connection for subsequent use.



Bus Phases Diagram

Information Transfer Phases

These six distinct phases are used to transfer information (e.g. command, data, status or message information) between the pair of devices to which the bus is allocated (e.g. the Initiator and the Target).

The following six information transfer phases are distinguished:

- Command Phase
- Data-In Phase
- Data-Out Phase
- Status Phase
- Message-In Phase
- Message-Out Phase

The bus can never be in more than one phase at any given time.

5.1.3 Signal specification

The bus has 18 signals divided into nine control signals and nine data signals.

The control signals are:

5.1.3.1 BUSY (BSY)

This signal may be driven by both the Initiator and the Target and indicates that the bus is being used.

5.1.3.2 SELECT (SEL)

This signal may be driven by the Initiator to select a Target or by a Target to reselect an Initiator.

5.1.3.3 REQUEST (REQ)

This signal shall be driven by the Target only to request a transfer to, or from, the Initiator during the information transfer phases.

5.1.3.4 ACKNOWLEDGE (ACK)

This signal shall be driven by the Initiator only to acknowledge a transfer to, or from, the Target during the information transfer phases.

5.1.3.5 CONTROL/DATA (C/D)

This signal shall be driven by the Target only, it indicates to the Initiator whether there is control or data information on the data bus during the information transfer phases.

C/D set to ZERO means data information.

C/D set to ONE means control information.

5.1.3.6 INPUT/OUTPUT (I/O)

This signal shall be driven by the Target only, it indicates to the Initiator the direction of information transfer during the information transfer phases.

I/O set to ZERO means output from the Initiator.

I/O set to ONE means input to the Initiator.

This signal is also used for connection/reconnection during the selection/reselection phase.

I/O set to ZERO means selection.

I/O set to ONE means reselection.

5.1.3.7 MESSAGE (MSG)

This signal shall be driven by the Target only, it indicates to the Initiator whether there is message or other information on the data bus during the information transfer phases.

MSG set to ZERO means command, data or status information.

MSG set to ONE means message information.

5.1.3.8 ATTENTION (ATN)

This signal shall be driven by the Initiator only, it indicates to the Target that the Initiator has a message ready.

5.1.3.9 RESET (RST)

This signal may be driven by any Bus Device, it indicates the reset condition to all bus devices.

The data signals are:

5.1.3.10 DATA BUS (DB1 to DB8, DBP)

The data bus has eight signals and a parity signal. The function of the data bus is dependent on the bus phase. Bit significance and signal number decrease downward from DB8 to DB1.

5.1.3.11 Arbitration, Selection and Reselection Phases

The data signals are uniquely assigned to each Bus Device and are used for Bus Device identification.

Data parity is odd. During the Arbitration phase parity is not valid and shall not be driven.

5.1.4 Signal Sequencing

The bus can never be in more than one phase at any given time. In the following descriptions, unless otherwise noted, signals which are not mentioned shall be released. Bus requesters are defined as Bus Devices that want to arbitrate for the bus, i.e. Initiators in the case of connection and Targets in the case of reconnection.

5.1.4.1 Bus Free Phase

This phase is defined by:

BSY set to ZERO
SEL set to ZERO

for a period not less than a Bus Settle delay.

5.1.4.2 Arbitration Phase

- Bus is in Bus Free phase

Enter Arbitration phase

- Bus Requester(s) : After detection of the Bus Free phase, and waiting a minimum of a Bus Free delay and a maximum of a Bus Set delay, they shall present their own Bus Device ID on DB and set BSY to ONE.

Inspect DB after waiting an Arbitration delay from the moment of setting BSY to ONE. If the priority of its own Bus Device ID is less than the priority of other (if any) Bus Device ID(s) on DB then release BSY and DB. If the priority is higher, set SEL to ONE.

Inspect SEL all the time during the Arbitration phase. If SEL is set to ONE (i.e. driven by the Bus Device that has won arbitration), then release BSY within a Bus Clear delay of SEL being set to ONE.

- The Bus Requester that wins arbitration shall wait a minimum of a Bus Clear delay plus a Bus Settle delay after setting SEL to ONE before changing any signals.
- During the Arbitration phase, parity is not used and DBP must be turned off or undriven by all bus devices.

The Bus is now allocated.

5.1.4.3 Selection Phase

Enter Selection Phase

- Initiator : Presents Target ID on DB and Initiator ID on DB. Set SEL to ONE and I/O to ZERO after waiting at least two Deskew delays from driving ID on DB. At least two Deskew delays later release BSY.
- All Bus Devices : Detect BSY set to ZERO, I/O set to ZERO and SEL set to ONE. Receive Target ID and Initiator ID on DB. Compare received Target ID with own ID. The selected Target is the one with the Target ID equal to its own ID.
- Selected Target : Set BSY to ONE within a Selection Abort delay of detection of being selected.
- Initiator : After releasing BSY, detects BSY set to ONE and SEL set to ONE. At least two Deskew delays later release SEL and DB.
- The Target(s) shall not respond (i.e. set BSY to ONE) if more than two device ID's are on the bus.

- Two alternative selection timeout procedures are specified for releasing the bus if the selected Target does not respond within a Selection Timeout delay.

  i) The Initiator sets RST to ONE.

  ii) The Initiator continues to set SEL to ONE and releases DB. If the Initiator does not detect BSY set to ONE within a Selection Abort delay plus two Deskew delays the Initiator shall allow the bus to enter the Bus Free phase by releasing SEL.

Exit Selection Phase

The Initiator is now connected to the Target.

5.1.4.4 Re-selection Phase

Enter Reselection Phase

- Target : Presents Initiator ID and Target ID on DB. Set SEL to ONE and I/O to ONE. After waiting at least two Deskew delays release BSY.

- All Bus Devices : Detect BSY set to ZERO, I/O set to ONE and SEL set to ONE. Receive Initiator ID and Target ID on DB. Compare received Initiator ID with own ID. The reselected Initiator is the one with the Initiator ID equal to its own ID.

- Reselected Initiator : Set BSY to ONE within a Selection Abort delay of detection of being selected.

- Target : After detecting BSY, set BSY to ONE and after waiting at least two Deskew delays release SEL.

- Reselected Initiator : Detects BSY set to ONE and SEL set to ZERO, then releases BSY.

The Initiator(s) shall not respond (i.e. set BSY to ONE) if more than two device IDs are on the bus.

Two alternative selection timeout procedures are specified for releasing the bus if the reselected Initiator does not respond within a Selection Timeout delay.

i) The Target sets RST to ONE.

ii) The Target continues to set SEL and I/O to ONE and releases DB. If the Target does not detect BSY set to ONE within a Selection Abort delay plus two Deskew delays the Target shall allow the bus to enter the Bus Free phase by releasing I/O and SEL.

Exit Re-selection Phase

The Target is now connected to the Initiator.

5.1.4.5 Information Transfer Phases

The six Information Transfer phases are distinguished by means of the three signals C/D, I/O and MSG; all driven by the Target. While within the Information Transfer phases the Initiator can request a Message-Out phase by setting ATN to ONE. While the Target can cause a Bus Free phase by releasing MSG, C/D, I/O and BSY.

Signal			Bus Phase	Direction of Information Transfer	
MSG	C/D	I/O			
0	0	0	Data-Out	Output	Data Phases
0	0	1	Data-In	Input	
0	1	0	Command	Output	
0	1	1	Status	Input	
1	0	0	These bit combinations are reserved for future standardization, they shall not be used		
1	0	1			
1	1	0	Message-Out	Output	Message Phases
1	1	1	Message-In	Input	

Input: To Initiator from Target

Output: From Initiator to Target

The same signal sequencing applied to all six phases and is only dependent on the direction of information transfer (i.e. input or output) and, for the data phases, on the data transfer mode (i.e. asynchronous or synchronous). The command, status and message phases use asynchronous mode only.

Input - Asynchronous Transfer Mode

(Bus phases: Data-In, Status and Message-In)

Sequence of events/signals:

- Initiator and Target are connected.

Enter Information Transfer Phase

- Target : Present type of information transfer phase on C/D, I/O and MSG. Wait at least a Bus Settle delay.

The following sequence of events/ signals is repeated for each byte of information.

m: Target : Present information on DB. After waiting at least one Deskew delay plus a Cable Skew delay set REQ to ONE.

n: Initiator : Detects REQ set to ONE. Receives type of information transfer phase on C/D, I/O and MSG. Set ACK to ONE after reception of the information on DB.

o: Target : Detects ACK set to ONE, then releases DB and REQ.

p: Initiator : Detects REQ set to ZERO, then releases ACK.

- Target : Detects ACK set to ZERO. If more information is to be transferred, repeat steps m, n, o, p.

- Initiator and Target are still connected.

Output - Asynchronous Transfer Mode

(Bus phases: Data-Out and Command)

- Initiator and Target are connected.

Enter Information Transfer Phase

- Target : Present type of information transfer phase on C/D, I/O and MSG. Wait at least a Bus Settle delay.

The following sequence of events/ signals is repeated for each byte of information.

m: Target : Set REQ to ONE.

n: Initiator : Detects REQ set to ONE. Receives type of information transfer phase on C/D, I/O and MSG. Present information on DB. After waiting at least one Deskew delay plus a Cable Skew delay set ACK to ONE.

o: Target : Detects ACK set to ONE. Receives information on DB and signals acceptance of information by releasing REQ.

p: Initiator : Detects REQ set to ONE, then releases ACK.

- Target : Detects ACK set to ZERO. If more information is to be transferred, repeat steps m, n, o, p.

- Initiator and Target are still connected.

Input - Synchronous

(Bus phase: Data-in)

Sequence of events/signals:

Enter Information Transfer Phase - Initiator and Target must have agreed previously via the OFFSET INTERLOCK DATA TRANSFER REQUEST message to use synchronous mode and they must have agreed a REQ/ACK offset and a Transfer Period delay. The REQ/ACK offset specifies the maximum number of REQ pulses which can be sent to the Target in advance of the number of ACK pulses received by the Initiator. The Transfer Period delay specifies the minimum time allowed between the leading edges of successive REQ pulses and of successive ACK pulses.

- Target : Presents data in phase on C/D, I/O and MSG. Wait at least a Bus Settle delay.

  The following sequence of events/ signals is repeated for each byte of information.

- Target : Presents information on DB. After waiting at least one Deskew delay plus a Cable Skew delay set REQ to ONE. Hold information on DB for a minimum of one Deskew delay plus one Hold Time delay after setting REQ to ONE.

  If more information is to be transferred repeat the above step.

- Initiator : Detects REQ pulse. Receives information on DB within one Hold Time delay of REQ being set to ONE.

- Initiator : Sends ACK pulse within REQ/ACK offset. Continues sending ACK pulses, separated by at least one Transfer Period delay until the number of ACK pulses sent equals the number of REQ pulses received.

- Target : Receives ACK pulses.

- Initiator and Target are still connected.

Output - Synchronous

(Bus phase: Data-out)

Sequence of events/signals:

- Initiator and Target are connected.

Enter Information Transfer Phase - Initiator and Target must have agreed previously via the OFFSET INTERLOCK DATA TRANSFER REQUEST message to use synchronous mode and they must have agreed a REQ/ACK offset and a Transfer Period delay.

- Target : Presents Data-out phase on C/D, I/O and MSG. Wait at least a Bus Settle delay.

  The following sequence of events/ signals is repeated for each byte of information.

- Target : Sends REQ pulse.

- Initiator : Detects REQ pulse. Receives type of information transfer phase on C/D, I/O and MSG.

- Initiator : Presents information on DB. After waiting at least one Deskew delay plus one Cable Skew delay, set ACK to ONE. Hold information on DB for a minimum of one Deskew delay plus one Hold Time delay after setting ACK to ONE. Release ACK.

  If more information is to be transferred repeat the above step.

  If no further information, release DB and continue.

- Target : Detects ACK pulse. Receives information on DB within one Hold Time delay of the transition of ACK to ONE.

- Initiator and Target are still connected.

Signal restrictions between phases

When the Bus is between two Information Transfer Phases, the following restrictions shall apply to the bus signals:

i) The BSY, SEL, REQ and ACK signals shall not change.

ii) The C/D, I/O, MSG and DATA BUS signals may change. When switching the direction of information transfer from out to in (Target to Initiator), the Target shall delay driving the DATA BUS signals by at least

a Data Release delay plus a Bus Settle delay after setting the I/O signal to ONE and the Initiator shall release the DATA BUS signals not later than a Data Release delay after the transition of the I/O signal to ONE. When switching the direction of information transfer from in to out (Initiator to Target), the Target shall release the DATA BUS not later than a Deskew delay after setting the I/O signal to ZERO.

iii) The ATN and RST signals may change as defined under the descriptions for the ATTENTION condition (5.1.5.1) and RESET condition (5.1.5.2).

5.1.5 Bus conditions

The Bus has two asynchronous conditions; the Attention condition and the Reset condition.

5.1.5.1 Attention condition

The Attention condition allows an Initiator to inform a Target that the Initiator has a message ready to transfer to the Target.

The Initiator creates the Attention condition by setting ATN to ONE at any time except during an Arbitration or a Bus Free phase. The Target may take the message at its convenience by going to the Message-Out phase before the next Bus Free phase occurs.

If more than one message byte is to be transferred, the Initiator shall continue to set ATN to ONE. The Initiator shall release ATN before setting ACK to ONE, following the request from the Target for the last data byte of the message to be transferred. The Target shall continue to receive message bytes until the Initiator releases ATN.

If the Target detects one or more parity errors in the message bytes received it may retry the message by setting REQ to ONE after detecting ATN set to ZERO prior to leaving the Message-Out phase. When re-transferring the message bytes the Initiator shall set ATN to ONE prior to setting ACK to ONE on the first message byte.

The Target indicates that it has received the message bytes successfully by changing from the Message-Out phase to any other Information Transfer phase and transferring at least one byte. Alternatively the Target may indicate that it has received the message successfully by changing to the Bus Free phase (e.g. following an ABORT or a RESET BUS DEVICE message).

5.1.5.2 Reset condition

The Reset condition is used to force all Bus Devices into the Bus Free phase. This condition takes precedence over all other phases and conditions. Any Bus Device may create the Reset condition by setting RST to ONE for a minimum of a Reset Hold Time delay. All Bus Devices shall release all Bus signals (except RST) within a Bus Clear delay of RST being set to ONE.

Upon receiving the Reset condition a Bus Device shall proceed to execute one of two courses of action dependent on whether Hard or Soft Reset is implemented.

5.1.5.2.1 Hard Reset

- Clear all uncompleted commands,
- release all Bus Device reservations,
- return any Bus Device operating modes (MODE SELECT, PREVENT/ALLOW MEDIUM REMOVAL commands, etc.) to their default condition.

5.1.5.2.2 Soft Reset

- Attempt to complete any uncompleted commands that were fully identified,
- preserve all Bus Device reservations,
- preserve any Bus Device operating modes (MODE SELECT, PREVENT/ALLOW MEDIUM REMOVAL commands, etc.).

The Soft Reset condition allows a single Initiator to reset the Bus without disturbing the operation of other Initiators in a multiple-Initiator system. To ensure proper operation the following conditions shall be met, as applicable:

i) An Initiator shall not consider a command to be fully identified until the IDENTIFY message is sent to the Target and the Target responds by changing to any other information transfer phase and requests that at least one byte be transferred.

ii) A Target shall consider a command to be fully identified when it successfully receives the IDENTIFY message.

iii) If an Initiator selects a Logical Unit for which there already is an active command for the same Initiator, the Target shall clear the original command and perform the new command.

iv) If a Target reselects an Initiator to continue a command for which the Initiator has no record, the Initiator shall abort that command by sending the ABORT message.

v) An Initiator shall consider a command to be completed when it does not set ACK for a successfully received COMMAND COMPLETED message.

vi) A Target shall consider a command to be completed when it detects the transition of ACK from ONE for the COMMAND COMPLETED message with ATN set to ZERO.

vii) An Initiator shall set ACK for the SAVE DATA POINTER message until it has actually saved the Data Pointer for the operation.

viii) A Target shall consider the Data Pointer to be saved when it detects the transition of ACK from ONE to ZERO for the SAVE DATA POINTER message with ATN set to ZERO.

ix) If the Reset condition occurs between the time that the Target sets REQ for the SAVE DATA POINTER message and it detects the transition of ACK from ONE to ZERO, the Target shall terminate the command with a CHECK CONDITION status. If Extended Sense Bytes are implemented, the Target shall set the Sense Key to ABORTED COMMAND. This is necessary because the Target cannot determine whether the Data Pointer has actually been saved.

NOTE 1

If the ATN signal is set to ONE in conditions vi) or viii), the Target would normally switch to the Message-Out phase and attempt to transfer a message byte. If the Reset condition occurs before it is able to successfully receive the message byte, the Target shall assume that the Initiator may not have successfully received the COMMAND COMPLETED message or the SAVE DATA POINTER message. In the case of the COMMAND COMPLETED message, the Target shall reselect the Initiator and attempt to send the COMMAND COMPLETED message again. In the case of the SAVE DATA POINTER message, the Target shall reselect the Initiator and terminate the command as described in condition ix).

5.1.6 Signal Timing

Arbitration Delay (2,2 us)

The minimum time a Bus Device shall wait from setting BSY to ONE for arbitration until the Data Bus can be examined to see if arbitration has been won. There is no maximum time.

Assertion Period Delay (90 ns)

The minimum time that a Target shall set REQ to ONE while using synchronous data transfer. Also, the minimum time that an Initiator shall set ACK to ONE while using synchronous data transfers.

Bus Clear Delay (88 ns)

The maximum time for a Bus Device to stop setting all bus signals to ONE after:

- Bus Free phase is detected (BSY and SEL both set to ZERO during a Bus Settle delay).
- SEL is received from another Bus Device during the Arbitration phase.

NOTE 2

For the first condition above, the maximum time for a Bus Device to clear the bus is 1200 ns from BSY and SEL first being set both to ZERO. If a Bus Device requires more than a Bus Settle delay to detect the Bus Free phase, it shall clear the bus within a Bus Clear delay minus the excess time.

Bus Free Delay (800 ns)

The minimum time that a Bus Device shall wait from its detection of the Bus Free phase (BSY and SEL both set to ZERO for a Bus Settle delay) until it sets BSY to ONE when going to the Arbitration phase.

Bus Set Delay (1,8 us)

The maximum time for a Bus Device to set BSY to ONE and its Bus Device ID bit to ONE on the Data Bus after it detects the Bus Free phase (BSY and SEL both set to ZERO for a Bus Settle delay) for the purpose of entering the Arbitration phase.

Bus Settle Delay (400 ns)

The time to wait for the bus to settle after changing certain control signals.

Cable Skew Delay (10 ns)

The maximum difference in propagation time allowed between any two Bus signals when measured between any two Bus Devices.

Data Release Delay (400 ns)

The maximum time for an Initiator to release the Data Bus signals following the transition of I/O from ZERO to ONE.

Deskew Delay (45 ns)

The minimum time required for deskew of certain signals.

Hold Time Delay (45 ns)

The minimum time added between setting REQ or ACK to ONE and the changing of the data lines to provide hold time in the Initiator or Target respectively, while using synchronous data transfers.

Non-assertion Period Delay (90 ns)

The minimum time that a Target shall set REQ to ZERO while using synchronous data transfers. Also, the minimum time that an Initiator shall set ACK to ZERO while using synchronous data transfers.

Reset Hold Time Delay (25 us)

The minimum time for which RST is set to ONE. There is no maximum time.

Selection Abort Time Delay (200 us)

The maximum time a Target (or Initiator) shall take from its most recent detection of being selected (or re-selected) until setting BSY to ONE. This timeout is required to ensure that a Target (or Initiator) does not set BSY to ONE after a Selection (or Re-selection) phase has been aborted.

Selection Timeout Delay (250 ms)

The minimum time that an Initiator (or Target) should wait for BSY set to ONE during the Selection (or Re-selection) phase before starting the timeout procedure.

Transfer Period Delay (set during a Message phase)

The minimum time allowed between the leading edges of successive REQ pulses and of successive ACK pulses while using Synchronous Data transfers.

5.2 Pointers

The Pointers contain the necessary information to allow a Target to perform a command. Each Initiator shall maintain a set of Pointers for each concurrent activity. These Pointers are specific to a given transfer and are:

- Command Pointer
- Data Pointer
- Status Pointer

A Pointer is an address, which identifies the storage location of the next byte to be transferred. Although maintained by the Initiator, a Pointer can be updated only after an action by the Target. Pointers are initially loaded by the Initiator at the commencement of each command, including each command within a sequence of linked commands. When the Target transfers a byte of information, the corresponding Pointer shall be incremented by the Initiator.

5.2.1 Current Pointers and Saved Pointers

Each Initiator shall have only one set of Pointers which are active, these are called the Current Pointers. It shall support a set of Saved Pointers for each concurrent

activity which it is capable of supporting. The Target, by sending the appropriate message, can cause the Initiator to save the current state of the Data Pointer. Conversely, the Target may ask the Initiator to restore the last saved set. If none was previously saved, the initial set shall be restored.

5.2.2 Command Pointer

The Command Pointer is an address which identifies the storage location of a command. After transfer of each byte of a command to the Target, the Pointer shall be incremented to the address of the next command byte. The Saved Command Pointer shall always point to the first byte of the current Command Descriptor Block.

5.2.3 Data Pointer

The Data Pointer is an address which identifies the storage location of the next byte of data to be transferred. After transfer of each data byte to, or from, the Target, the Pointer shall be incremented to the address of the next byte. The Target can request the Initiator to save the Current Data Pointer by use of the SAVE DATA POINTER message.

5.2.4 Status Pointer

The Status Pointer is an address which identifies the storage location of the Status Byte stored by the Target on completion of a command. The Saved Status Pointer shall always point to the Status Byte for the current command.

5.2.5 Restoring the Pointers

The Target by use of the RESTORE POINTERS message causes the Initiator to restore the most recently saved Pointers (for the currently attached Logical Unit) to the active state. Command and Status Pointers shall be restored to their original value at the beginning of the current command. The Data Pointer shall be restored to the point at which the last SAVE DATA POINTER message occurred for that Logical Unit, or, failing this message, to its initial value.

5.3 Disconnection and Reconnection

After a SELECTION phase or an INFORMATION TRANSFER phase a Target may wish to be disconnected from the Initiator, to perform some time-consuming task without holding up the bus. This allows the bus to revert to the BUS FREE phase. When the Target wishes to continue the operation, it must apply for a RESELECTION phase.

5.3.1 Disconnection

DISCONNECT is a message sent to the Initiator by a Target which wishes to relinquish control of the bus for a period of time. A later RECONNECTION phase will be required to complete the current activity. The DISCONNECT message will not save the Current Data Pointer. If the Target requires it to be saved it must first issue a SAVE DATA POINTER message before the DISCONNECT message.

5.3.2 Reconnection

The Target requests the bus by entering an ARBITRATION phase, upon obtaining the bus the Target, after a RESELECTION phase, is connected again with the Initiator and sends the IDENTIFY message. This action will cause the Initiator to restore the last set of Saved Pointers for the LUN (if no Pointer has been previously saved, then Initial Pointers shall be restored). The current activity can then continue.

5.4 Messages

The message system allows communication between an Initiator and Target for the physical path management.

All Bus Devices shall implement the COMMAND COMPLETED message and all Bus Devices that implement any other message shall also implement the MESSAGE REJECTED message. The Bus Devices indicate their ability to accomodate more than the COMMAND COMPLETED message by activating or responding to the ATN signal.

The Initiator indicates this by activating the ATN signal prior to the bus state with SEL set to ONE and BSY set to ZERO in the SELECTION phase.

The Target indicates its ability to accommodate more messages by responding to the ATN signal with the MESSAGE OUT phase after going through the SELECTION phase.

The first message sent by the Initiator after the SELECTION phase shall be the IDENTIFY message. This permits the establishment of the physical path for a particular Logical Unit specified by the Initiator.

After the RESELECTION phase, the Target's first message shall be the IDENTIFY message. This permits the physical path to be re-established for the LUN specified in the message.

If an error condition occurs, an Initiator may send the ABORT message or the BUS DEVICE RESET message instead of the IDENTIFY message, as the first message. Only one LUN may be identified for any one selection and a second IDENTIFY message with a new LUN may not be issued before the bus has been released.

Whenever a physical path is established in an Initiator that can accommodate the DISCONNECT message and the RESELECTION phase, the Initiator must assure that the active Pointers of the physical path are equal to the Saved Pointers for that particular Logical Unit.

5.4.1 Message Format

This Standard specifies two message formats:

- Non-extended messages
- Extended messages.

5.4.1.1 Non-extended messages

Non-extended messages shall consist of one single 8-bit byte identifying it.

5.4.1.2 Extended messages

Extended messages shall consist of several bytes set as follows:

The 1st byte	shall be set to (01), indicating that the message is an extended message.
The 2nd byte	shall be set to the number of bytes following it. (00) shall indicate that 256 bytes are following.
The 3rd byte	shall identify the extended message.
The 4th to nth byte	shall specify the message arguments.

5.4.2 Message Coding

Message Name		Source	Number of bytes	First or unique byte	3rd byte
COMMAND COMPLETED	B	T	1	(00)	-
SAVE DATA POINTER	Nm	T	1	(02)	-
RESTORE POINTERS	Nm	T	1	(03)	-
DISCONNECT	E	T	1	(04)	-
LINKED COMMAND COMPLETED	Nm	T	1	(0A)	-
LINKED COMMAND COMPLETED WITH FLAG	Nm	T	1	(0B)	-
MODIFY DATA POINTER*	Nm	T	7	(01)	(00)
INITIATOR DETECTED ERROR	Nm	I	1	(05)	-
ABORT	Nm	I	1	(06)	-
NO OPERATION	Nm	I	1	(08)	-
MESSAGE PARITY ERROR	Nm	I	1	(09)	-
RESET BUS DEVICE	Nm	I	1	(0C)	-
MESSAGE REJECTED	E	T,I	1	(07)	-
IDENTIFY	E	T,I	1	(80)-(87) (C0)-(C7)	-
EXTENDED IDENTIFY*	Nm	T,I	4	(01)	(02)
OFFSET INTERLOCK DATA TRANSFER REQUEST*	Nm	T,I	5	(01)	(01)

T = Target
I = Initiator
* = Extended message

B = mandatory in Basic Level
E = mandatory in Extended Level
Nm = not mandatory in either level

5.4.3 Messages from Target

5.4.3.1 COMMAND COMPLETED (00)

This message indicates that the execution of a command (or a series of linked commands) has been terminated and that status has been sent to the Initiator. After successfully sending this message the Target shall release the bus.

The command may have been executed successfully or unsuccessfully as indicated in the status.

5.4.3.2 SAVE DATA POINTER (02)

This message requires the Initiator to save a copy of the present active Data Pointer for the currently attached Logical Unit.

5.4.3.3 RESTORE POINTERS (03)

This message requires the Initiator to restore the most recently saved Pointers (for the currently attached LUN) to the active Pointers. Command and Status Pointers shall be restored to the beginning of the current command. The Data Pointer shall be restored to the beginning of the data area in the absence of a SAVE DATA POINTER message or to the point at which the last SAVE DATA POINTER message occurred for that LUN.

5.4.3.4 DISCONNECT (04)

This message informs the Initiator that the present physical path is going to be broken (the Target will release the bus), but that a later reconnection will be required in order to complete the current operation. If the Initiator detects the BUS FREE phase (other than as a result of the RESET BUS DEVICE message, the ABORT message or a Hard Reset condition) without first receiving a DISCONNECT message or a COMMAND COMPLETED message the Initiator shall consider this as a fatal error. If the Target intentionally creates this condition it shall abort the current command.

5.4.3.5 LINKED COMMAND COMPLETED (0A)

This message indicates that the execution of a linked command has been completed successfully and that status information has been sent. The Initiator is then permitted to set up the Pointers for the initial state for the next linked command.

5.4.3.6 LINKED COMMAND COMPLETED WITH FLAG (0B)

This message indicates that the execution of a linked command has been completed successfully and that status information has been sent.

The Initiator is then permitted to set up the Pointers for the initial state for the next linked command.

5.4.3.7 MODIFY DATA POINTER (Extended message)

This message requires that the signed argument be added (two's complement) to the present value of the Data Pointer.

This message shall comprise 7 bytes set as follows:

The 1st byte	shall be set to (01).
The 2nd byte	shall be set to (05).
The 3rd byte	shall be set to (00).
The 4th byte	shall be set to the most-significant byte of the argument.
The 5th byte	shall be set to the next byte of the argument.
The 6th byte	shall be set to the next byte of the argument.
The 7th byte	shall be set to the least-significant byte of the argument.

5.4.4 Messages from Initiator

5.4.4.1 INITIATOR DETECTED ERROR (05)

This message indicates that the Initiator has detected an error (normally a data parity error) since the last time the state of the Data Pointer was saved. As a retry procedure the Target may repeat the same operation by sending the RESTORE POINTERS message or the DISCONNECT message followed by a reconnect.

5.4.4.2 ABORT (06)

This message requires the Target to clear the current operation for the specified LUN from the selecting Initiator and cause the Target to release the bus. No status or ending message shall be sent for the aborted operations. Only the operations from the particular Initiator are affected. If no LUN has been selected by the IDENTIFY message, then all operations for the selected Target from that Initiator will be aborted.

It is not an error to issue this message to an LUN that is not currently performing an operation for the Initiator.

5.4.4.3 NO OPERATION (08)

This message shall be sent from an Initiator in response to a request of the Target for a message when the Initiator does not currently have any other valid message to send.

5.4.4.4 RESET BUS DEVICE (0C)

This message requires the Target to go to an initial state with no operations pending for any Initiator and releases the bus. No status or ending messages shall be sent for the operations aborted.

5.4.4.5 MESSAGE PARITY ERROR (09)

This message indicates that one or more bytes in the last message received had a parity error.

In order to indicate its intention of sending this message, the Initiator shall drive the ATN signal prior to its release of ACK for the REQ/ACK handshake of the message that had a parity error. This provides an interlock so that the Target can determine which message has the parity error.

5.4.5 Messages from Initiator or Target

5.4.5.1 MESSAGE REJECTED (07)

This message indicates that the last message was inappropriate or has not been implemented.

In order to indicate its intention of sending this message, the Initiator shall drive the ATN signal prior to its release of ACK for the REQ/ACK handshake of the message that will be rejected.

When a Target sends this message, it shall change to MESSAGE IN phase and send this message prior to requesting additional message bytes from the Initiator. This provides an interlock so that the Initiator can determine which message is rejected.

5.4.5.2 IDENTIFY (08) to (87) and (C0) to (C7)

This message shall establish the physical path connection between an Initiator and Target for a particular LUN.

Bit 8	shall be set to ONE (to distinguish this message from others).
Bit 7	This bit is set by the Initiator. When set to ONE it indicates that the Initiator has the ability to accommodate disconnection and reconnection.
Bits 6 - 4	shall be set to ZERO.
Bits 3 - 1	shall indicate, in binary notation, the address of the Logical Unit (LUN) in a Target.

When this message is sent from a Target during reconnection the Initiator shall also restore Pointers as if it has received a RESTORE POINTERS message.

5.4.5.3 OFFSET INTERLOCK DATA TRANSFER REQUEST (Extended message)

This message is used to establish the synchronous mode of data transfer between two Bus Devices and defines a minimum period for the REQ and ACK signals and a maximum offset permitted between a REQ signal and its corresponding ACK signal.

A pair of OFFSET INTERLOCK DATA TRANSFER REQUEST messages are exchanged between an Initiator and a Target whenever the Bus Device that can support synchronous data transfer recognizes that it has not communicated with the other Bus Device since receiving the last BUS DEVICE RESET message or a HARD RESET condition.

The Bus Device may also exchange messages to establish synchronous data transfer when requested to do so. The message exchange establishes the transfer period and the REQ/ACK offset. The transfer period is the minimum time between leading edges of successive REQ and of successive ACK pulses. The REQ/ACK offset is the maximum number of REQ pulses that may be outstanding before its corresponding ACK pulse is received.

If the Initiator is the first to detect that it must send an OFFSET INTERLOCK DATA TRANSFER REQUEST message it shall send the message indicating an REQ/ACK offset and minimum transfer period. The REQ/ACK offset is chosen to prevent Initiator buffer overflows, while the minimum transfer period is chosen to meet the data handling requirements of the Initiator. The Target responds in any of the following ways.

Target response	Implied agreement
REQ/ACK offset less than, or equal to, requested value	REQ/ACK offset equal to Target value.
Minimum transfer period is greater than, or equal to, requested period.	Minimum transfer period equal to Target value.
REQ/ACK offset equal to zero.	Asynchronous transfer
MESSAGE REJECTED message.	Asynchronous transfer.

If the Target recognizes that re-negotiation is required, it sends an OFFSET INTERLOCK DATA REQUEST message to the Initiator. The REQ/ACK offset is selected to prevent buffer and offset counter overflows, while the minimum transfer period is chosen to meet the data handling requirements of the Target.

The Initiator responds in any of the following ways if the Target selects an REQ/ACK offset equal to (FF).

Initiator response	Implied agreement
REQ/ACK offset equal to (FF).	REQ/ACK offset unlimited.
Minimum transfer period equal to, or greater than, requested period.	Minimum transfer period equal to the Target value.
REQ/ACK offset equal to zero.	Asynchronous transfer.
MESSAGE REJECTED message.	Asynchronous transfer.

The Initiator responds in any of the following ways if the Target requests an offset less than (FF).

Initiator response	Implied agreement
REQ/ACK offset less than, or equal to, the requested value.	REQ/ACK offset equal to the Initiator value.
Minimum transfer period equal to, or greater than, requested value.	Minimum transfer period equal to the Initiator value.
REQ/ACK offset equal to zero.	Asynchronous transfer.
MESSAGE REJECTED message.	Asynchronous transfer.

The implied agreement between two devices shall remain in effect until a RESET BUS DEVICE message or a Hard Reset condition is received or until one of the two Bus Devices is required by external conditions to modify the agreement.

The default mode of data transfer is asynchronous mode. The default mode is entered with power on, after a RESET BUS DEVICE message or a Hard Reset condition.

The OFFSET INTERLOCK DATA TRANSFER REQUEST message exchange can only take place immediately following a SELECTION phase that includes both the Initiator ID and the Target ID.

Violation of this rule may make data transfer impossible due to device disagreements about the data transfer mode.

This message shall comprise 5 bytes set as follows:

The 1st byte shall be set to (01).

The 2nd byte shall be set to (03).

The 3rd byte shall be set to (01).

The 4th byte shall indicate in binary notation 1/4 of the duration of the transfer period in ns.

The 5th byte shall indicate the REQ/ACK offset. If set to (00) it indicates asynchronous mode, if set to (FF) it indicates unlimited offset.

5.4.5.4 EXTENDED IDENTIFY (extended message)

This message may be used in conjunction with the normal IDENTIFY message in order to expand the LUN address space in a Target. Each LUN may be divided into 256 SUB-LUN addresses. This allows up to 2048 units to be addressed in one Target.

This message shall comprise 4 bytes set as follows:

The 1st byte shall be set to (01).

The 2nd byte shall be set to (02).

The 3rd byte shall be set to (02).

The 4th byte shall specify the SUB-LUN address from 1 (= (00)) to 256 (= (FF)).

SECTION IV

6. SPECIFICATION OF THE SET OF COMMANDS

6. SPECIFICATION OF THE SET OF COMMANDS

The command structure, using a logical block addressing method rather than a physical addressing method, provides the host system with device independance within a class of devices.

A command may be unique to a device, or may have interpretations, fields, or features which are specific to a device type.

A command is specified by a set of bytes called: Command Descriptor Block (CDB).

Some commands may require additional descriptive information in the form of a Parameter List sent as data.

Unless otherwise specified, all fields of the CDB or of the Parameter List are recorded in binary notation.

The meaning of a number of single bits, of bytes and of whole fields is not specified. They are reserved for future standardization. They shall be set to ZERO. A Target that receives a single bit, a byte or a field that is set to another value than the mandatory ZERO shall terminate the command with a CHECK CONDITION status. If Extended Sense is implemented, the Sense Key shall be set to ILLEGAL REQUEST.

Write or read type commands allow the transfer of one or more logical blocks of data.

Information in the CDB allows the linking between commands to the same Logical Unit.

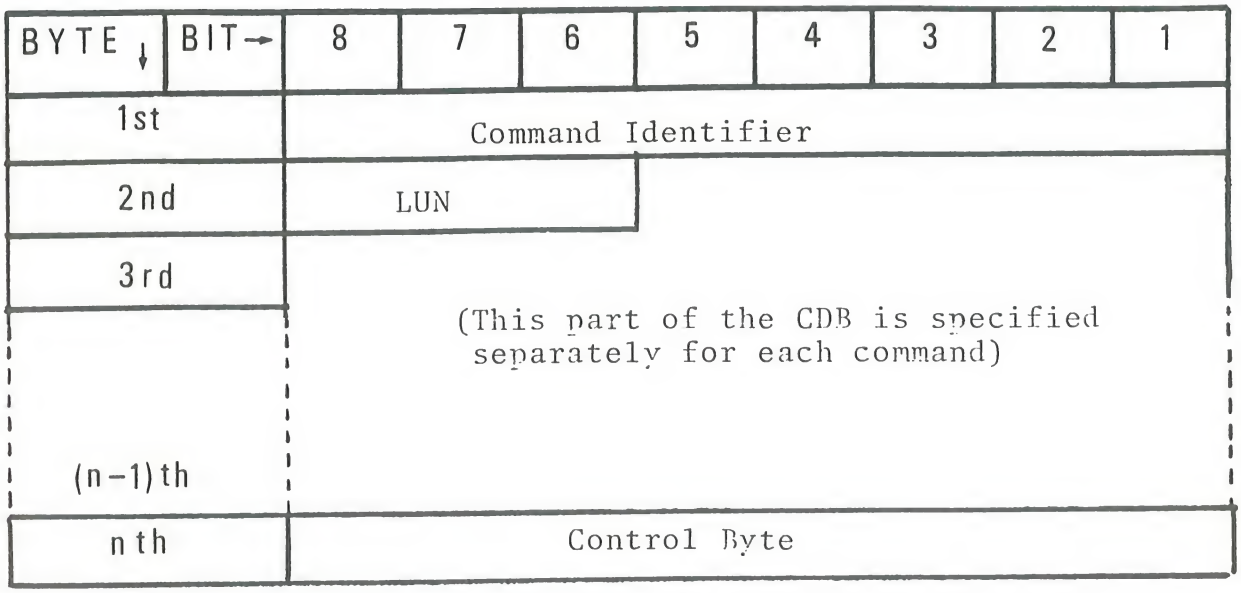
After a command phase a Target may disconnect itself from the bus to allow activity by other Bus Devices while a Logical Unit is preparing for data transfer.

After each command execution (successful or unsuccessful) the Target shall send to the Initiator an ending status and one of the three possible messages indicating that a command has been completed.

When an ending status indicates an unsuccessful completion of the command, the Initiator may send the REQUEST SENSE command for additional information in order to identify the reason for the abnormal command completion.

6.1 Definition of the Command Descriptor Block (CDB)

The general format of a CDB shall be:



1st byte

The first byte of the CDB is called Command Identifier. The 256 possible commands are grouped in eight groups of 32 commands each.

Bits 8, 7, 6 specify the group, bits 5, 4, 3, 2, 1 specify the individual command within the group.

2nd byte

Bits 8, 7, 6 specify the Logical Unit Number.

Bits 5 to 1 are dependent on each specific command.

3rd to (n-1)th bytes

These bytes, together with bits 5 to 1 of the 2nd byte, are dependent on each specific command.

n-th byte

This byte is a control byte.

6.1.1 Command Groups

In this version of the Standard only two groups of commands are fully defined, others are reserved for future standardization, others are left for private use and therefore not specified at all.

Group 0 comprises all commands with a CDB of 6 bytes. This group is fully defined, whereby some commands are left for private use.

Group 1 comprises all commands with a CDB of 10 bytes. This group is fully defined.

Groups 2, 3 and 4 are reserved for future standardization. They shall not be used.

Group 5 comprises all commands with a CDB of 12 bytes. Commands identified by (0F) to (1F) are reserved for future standardization and shall not be used. Commands identified by (00) to (0E) are left for private use and are not specified.

Groups 6 and 7 are left for private use and are not specified.

Summary of Command Groups

Identifiers of the commands	Groups							
	0	1	2	3	4	5	6	7
(00)	Nm	PU	Shall not be used	Shall not be used	Shall not be used	Private Use	Private Use	Private Use
(01)	D	PU						
(02)	PU	PU						
(03)	B	PU						
(04)	D	PU						
(05)	D	D						
(06)	PU	PU						
(07)	D	PU						
(08)	D	D						
(09)	PU	PU						
(0A)	D	D						
(0B)	D	PU						
(0C)	PU	PU						
(0D)	PU	PU						
(0E)	PU	D						
(0F)	D	D						
(10)	D	D				Shall not be used		
(11)	D	D						
(12)	E	D						
(13)	D	D						
(14)	D	R						
(15)	D	R						
(16)	D	R						
(17)	D	R						
(18)	Nm	R						
(19)	D	Nm						
(1A)	D	Nm						
(1B)	D	R						
(1C)	Nm	R						
(1D)	Nm	R						
(1E)	D	R						
(1F)	R	R						

B : mandatory in Basic Level
E : mandatory in Extended Level
Nm: not mandatory
D : device-dependent (see each device type)
R : reserved for future standardization; shall not be used
PU: private use, not specified by this Standard

6.1.2 Logical Unit Number

The three higher bits of the 2nd byte of the CDB, i.e. bits 8, 7, 6, indicate one of the eight available Logical Unit Numbers (LUN).

Up to eight devices may be handled by a Target device.

This addressing method is used only for systems that do not implement the IDENTIFY message. When it is implemented the Target device receives the LUN through the IDENTIFY message and bits 8, 7, 6 of the 2nd byte of the CDB are ignored.

6.1.3 Command-dependent Information

The meaning of bits 5 to 1 of byte 2 and of bytes 3 to (n-1) is specified for each command, they are described separately.

For the commands that are transferring data they mainly define the logical block address of the first block of data to be transferred and the transfer length, i.e. the number of blocks to be transferred as indicated for each command.

6.1.4 Control Byte

The last byte of the CDB is called Control Byte.

Bit 1 : Link Bit

- Set to ZERO — shall mean that a NORMAL COMPLETION status shall be sent after the current command.
- Set to ONE — shall mean an automatik link to the next command upon successful completion of the current command.

  Targets supporting linked commands shall, upon successful completion of the current command, return an INTERMEDIATE status sent to the host and shall send one of the two messages specified by the Flag bit.

  Targets not supporting linked commands shall return a CHECK CONDITION status and set the Sense Key to ILLEGAL REQUEST in the Extended Sense Bytes if they are implemented.

Bit 2 : Flag bit

If the Link Bit is set to ZERO, the Flag Bit shall be ignored.

If the Link Bit is set to ONE the Flag Bit shall have the following meaning.

- Set to ZERO shall mean that the Target shall send a LINKED COMMAND COMPLETED message to the Initiator after an INTERMEDIATE status has been sent.

- Set to ONE shall mean that the Target shall send a LINKED COMMAND COMPLETED with a FLAG message to the Initiator after an INTERMEDIATE status has been sent.

Bits 3 to 6

These bits are reserved for future standardization and shall be set to ZERO.

Bits 7 and 8

These bits are left for private use, their meaning is not specified by this Standard.

6.2 Commands Common for all Devices

The following commands are applicable to all devices.

Name		Coded Representation
COMPARE	Nm	(39)
COPY	Nm	(18)
COPY AND VERIFY	Nm	(3A)
INQUIRY	E	(12)
RECEIVE DIAGNOSTIC RESULTS	Nm	(1C)
REQUEST SENSE	B	(03)
SEND DIAGNOSTIC	Nm	(10)
TEST UNIT READY	Nm	(00)

B = mandatory command in Basic Level

F = mandatory command in Extended Level

Nm= not mandatory in either level

Each command is described in detail in the following clauses.

6.2.1 COMPARE Command (39)

Common command.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	1	1	0	0	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	Parameter List Length (MSB)							
5th	Parameter List Length							
6th	Parameter List Length (LSB)							
7th	0	0	0	0	0	0	0	0
8th	0	0	0	0	0	0	0	0
9th	0	0	0	0	0	0	0	0
10th	Private Use		0	0	0	0	Flag	Link

The COMPARE command provides a means to compare data from one Logical Unit with those of another or of the same Logical Unit in a manner similar to the COPY command (see 6.2.2).

This command functions in the same manner as the COPY command, except that the data from the source is compared on a byte-by-byte basis with the data from the destination.

See the COPY command for additional information about this command.

If the comparison is unsuccessful the command shall be terminated with a CHECK CONDITION status and the Sense Key set to MISCOMPARE.

6.2.2 COPY Command (18)

Common command.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	1	0	0	0
2nd		LUN			0	0	0	0	0
3rd		Parameter List Length (MSB)							
4th		Parameter List Length							
5th		Parameter List Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The COPY command provides a means to copy data from one Logical Unit to another or to the same Logical Unit. The Logical Units may reside on the same Bus Device or different Bus Devices. Some Bus Devices which implement this command may not support copies to/from another Bus Device or third party copies (both Logical Units reside on other Bus Devices).

The Parameter List Length field specifies the length in bytes of the Parameter List, specifying the parameters which are sent during the DATA OUT phase of the command. If the value specified by the Parameter List Length field equals zero no data shall be transferred. This shall not be considered an error condition.

Write-Once-Multiple-Read Devices and Read-Only-Direct-Access Devices are considered to be Direct Access Devices for the COPY command. Printer Devices are considered to be Sequential Access Devices for the COPY command.

The Parameter List begins with a four-byte Header which contains the identifiers of the COPY functions and their priority. Following the Header is one or more Segment Descriptors. The number of parameters sent is equal to the value specified by the Parameter List Length field minus 4.

Parameter List

Header

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	COPY Function					Priority		
2nd	Private Use							
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0

Segment Descriptor(s)

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st								
n th								

The COPY functions define each a specific format for the Segment Descriptors. The COPY functions are coded as follows.

Name of the COPY Function	Coded Representation
Direct-Access-to-Sequential-Access	(00)
Sequential-Access-to-direct-Access	(01)
Direct-Access-to-Direct-Access	(02)
Sequential-Access-to-Sequential-Access	(03)
(Reserved for future standardization)	(04) to (0F)
(Private Use)	(10) to (1F)

Bits 1 to 3 of the 1st byte of the Header, the Priority field, establishes the relative priority of this COPY command to other commands being executed by the same bus device. All other commands are assumed to have a priority of 1. Priority 0 is the highest priority with increasing values indicating lower priorities.

The Segment Descriptor formats are determined by the type of COPY function. These formats are described in clauses 6.2.3.3 to 6.2.3.6. A maximum of 256 Segment Descriptors are permitted. The Segment Descriptors are identified by ascending numbers beginning with zero.

6.2.2.1 Errors Detected by the Managing Bus Device

Two classes of unusual conditions may occur during execution of a COPY command. The first class consists of those unusual conditions detected by the Bus Device that received the COPY command and is managing the execution of the command. These conditions include parity errors while transferring COPY command and Status Bytes, invalid parameters in the COPY command, invalid Segment Descriptors, and inability of the Bus Device controlling the COPY Functions to continue operating. In the event of such an unusual condition, the Bus Device managing the COPY operation shall:

- Terminate with CHECK CONDITION status.
- The Sense Data shall be returned in the Extended Sense format. The Validity bit shall be set to ONE. The Segment Number field shall contain the number of the Segment Descriptor being processed at the time the unusual condition is detected. The Sense Key shall be that describing the unusual condition. The Information Bytes field shall contain the difference between the value specified by the Number of Blocks field in the Segment Descriptor being processed at the time of the failure and the number of blocks successfully copied. This difference is the number of unprocessed blocks remaining for the Segment Descriptor.

6.2.2.2 Errors Detected by the Target

The second class of errors consists of unusual conditions detected by the Bus Device transferring data at the request of the Bus Device managing the transfer. The Bus Device managing the COPY command detects unusual conditions by receiving a CHECK CONDITION status from one of the Bus Devices it is managing. It then must recover the Sense Data associated with the unusual condition.

The Bus Device managing the COPY command may also be the Source or Destination Bus Device (or both). It must distinguish between a failure of the management of the copy operation and a failure of the data transfer being requested.

It shall then create the appropriate Sense Data internally.

After recovering the Sense Data associated with the detected error, the Bus Device managing the COPY command shall:

SEGMENT DESCRIPTOR

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Source Address			0	0	Source LUN		
2nd	Destination Address			0	0	Destination LUN		
3rd	Sequential Access Device Block Length (MSB)							
4th	Sequential Access Device Block Length (LSB)							
5th	Direct Access Device Number of Blocks (MSB)							
6th	Direct Access Device Number of Blocks							
7th	Direct Access Device Number of Blocks							
8th	Direct Access Device Number of Blocks (LSB)							
9th	Direct Access Device Logical Block Address (MSB)							
10th	Direct Access Device Logical Block Address							
11th	Direct Access Device Logical Block Address							
12th	Direct Access Device Logical Block Address (LSB)							

Source Address and Destination Address fields specify the Bus Devices and the Source LUN and Destination LUN fields specify the Logical Units to use for the COPY operation. Some Bus Devices may not support third party copy where the copying Bus Device is not the source or destination device. Some Bus Devices only support copy within the Bus Device and not tother other Bus Devices. If an unsupported COPY operation is requested the command shall be terminated with CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

The Sequential Access Device Block Length field specifies the Block Length to be used on the Sequential Access Logical Unit during the copy operation. If this block length is known by the Bus Device managing the copy to be not supported, the command shall be terminated with CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST. If the block length is found to be invalid while executing a read or write operation to the Sequential Access Device, the command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set to COPY ABORTED.

The Direct Access Device Number of Blocks field specifies the number of blocks in the current segment. A value of zero indicates that no blocks are to be transferred in this segment.

- Terminate the COPY command with CHECK CONDITION status.
- The Sense Data shall be returned in the Extended Sense format. The Validity bit shall be set to ONE. The Segment Number field shall contain the number of the Segment Descriptor being processed at the time the unusual condition is detected. The Sense Key shall be set to COPY ABORTED. The Information Bytes field shall contain the difference between the value specified by the Number of Blocks field in the Segment Descriptor being processed at the time of the failure and the number of blocks successfully copied. This difference is the number of unprocessed blocks remaining for the Segment Descriptor. The Additional Sense Length field shall specify the number of Additional Sense Bytes.

The first Additional Sense Byte specifies the byte number, relative to the first byte of Sense Data of the beginning of the Source Logical Unit's Status Byte and Sense Data. A zero value shall indicate that no Status Byte or Sense Data is being returned for the Source Logical Unit. The first byte of the area to which the first Additional Sense Byte points shall contain the Status Byte from the Source Logical Unit. The subsequent bytes shall contain, unchanged, the Sense Data recovered from the Source Logical Unit.

The second Additional Sense Byte specifies the byte number, relative to the first byte of Sense Data of the beginning of the Destination Logical Unit's Status Byte and Sense Data. A zero value shall indicate that no Status Byte or Sense Data is being returned for the Destination Logical Unit. The first byte of the area to which the second Additional Sense Byte points shall contain the Status Byte from the Destination Logical Unit. The subsequent bytes shall contain, unchanged, the Sense Data recovered from the Destination Logical Unit.

6.2.2.3 Copy Functions Direct-Access-to-Sequential-Access (00) and Sequential-Access-to-Direct-Access (01)

The format for the Segment Descriptors for COPY transfers between Direct Access and Sequential Access devices is specified in the table below. This format is required for COPY Functions Direct-Access-to-Sequential-Access and Sequential-Access-to-Direct-Access. The Segment Descriptor may be repeated up to 256 times within the Parameter List Length specified in the Command Descriptor Block.

The Direct Access Device Logical Block Address field specifies the starting Logical Block Address on the Logical Unit for this segment.

6.2.2.4 COPY Function Direct-Access-to-Direct-Access (02)

The format for the Segment Descriptors for copy transfers among Direct Access Devices is specified in the table below. This format is required for Copy Function Direct-Access-to-Direct-Access. The Segment Descriptor may be repeated up to 256 times within the Parameter List Length specified in the Command Descriptor Block.

SEGMENT DESCRIPTOR

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Source Address			0	0	Source LUN		
2nd	Destination Address			0	0	Destination LUN		
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Source Number of Blocks (MSB)							
6th	Source Number of Blocks							
7th	Source Number of Blocks							
8th	Source Number of Blocks (LSB)							
9th	Source Logical Block Address (MSB)							
10th	Source Logical Block Address							
11th	Source Logical Block Address							
12th	Source Logical Block Address (LSB)							
13th	Destination Logical Block Address (MSB)							
14th	Destination Logical Block Address							
15th	Destination Logical Block Address							
16th	Destination Logical Block Address (LSB)							

The Source Address and Destination Address fields specify the Bus Devices and the Source LUN and Destination LUN fields specify the Logical Units to use for the copy operation. Some Bus Devices may not support third party copy where the copying Bus Device is not the source or destination device. Some Bus Devices only support copy within the Bus Device and not other Bus Devices. If an unsupported copy operation is requested the command shall be terminated with CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

The Source Number of Blocks field specifies the number of blocks to be transferred from the source device during command execution.

The Source Logical Block Address field specifies the starting logical block address on the source device.

The Destination Logical Block Address field specifies the starting logical block address on the destination device.

6.2.2.5 COPY Function Sequential-Access-to-Sequential-Access (03)

The format for the Segment Descriptors for copy transfers among Sequential Access devices is specified by the table below. This format is required for Copy Function Sequential-Access-to-Sequential-Access. The Segment Descriptor may be repeated up to 256 times within the Parameter List Length specified in the Command Descriptor Block.

SEGMENT DESCRIPTOR

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Source Address			0	0	Source LUN		
2nd	Destination Address			0	0	Destination LUN		
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Source Block Length (MSB)							
6th	Source Block Length							
7th	Destination Block Length (MSB)							
8th	Destination Block Length (LSB)							
9th	Source Number of Blocks (MSB)							
10th	Source Number of Blocks							
11th	Source Number of Blocks							
12th	Source Number of Blocks (LSB)							

Source Address and Destination Address fields specify the Bus Devices and the Source LUN and Destination LUN fields specify the Logical Units to use for the copy operation. Some Bus Devices may not support third party copy where the copying Bus Device is not the source or destination device. Some Bus Devices only support copy within the Bus Device and not to other Bus Devices.

If an unsupported copy operation is requested the command shall be terminated with CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

The Source Block Length field specifies the block length of the source device for this segment of the copy. ZEROs in this field indicate variable block length. For non-zero values, the value specified by this field shall be equal to the Logical Unit's actual block length. If block length inequalities are detected by the Bus Device managing the copy, the command shall be terminated with CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST. If the inequalities are detected during the read operation by the copy manager, the command shall terminate with a CHECK CONDITION status and the Sense Key set to COPY ABORTED.

The Destination Block Length field specifies the block length to be used on the destination Logical Unit during the copy. Inequalities between the block length specified by the Destination Block Length field and that of the block in the destination LUN are handled in the same manner as specified for the Source Block Length field.

The Source Number of Blocks field specifies the number of blocks to be transferred from the source device during this segment. A value of zero means that no blocks are to be transferred.

6.2.3 COPY AND VERIFY Command (3A)

Common command.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	1	1	0	1	0
2nd	LUN			0	0	0	BC-Bit	0
3rd	0	0	0	0	0	0	0	0
4th	Parameter List Length (MSB)							
5th	Parameter List Length							
6th	Parameter List Length (LSB)							
7th	0	0	0	0	0	0	0	0
8th	0	0	0	0	0	0	0	0
9th	0	0	0	0	0	0	0	0
10th	Private Use		0	0	0	0	Flag	Link

The COPY AND VERIFY command performs the same function as the COPY command, except that a verification of the data written to the Destination Logical Unit is performed after the data is written. See the COPY command for additional information about this command.

The Byte Check Bit (BC-Bit), if set to ZERO, causes the verification to be simply a medium verification (CRC, ECC, etc.). The Byte Check Bit, if set to ONE, causes a comparison byte-by-byte of data written to the Destination Logical Unit Number and the data read from the Source Logical Unit. If the comparison is unsuccessful the command shall be terminated with a CHECK CONDITION status and the Sense Key set to MISCOMPARE.

6.2.4 INQUIRY Command (12)

Common command.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	0	1	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Number of Allocated Bytes							
6th	Private Use		0	0	0	0	Flag	Link

This command allows the Initiator to receive parameters related to the Target and/or to the addressed Logical Unit. The 5th byte, the Number of Allocated Bytes field, specifies the number of Inquiry Data bytes allocated to be sent to the Initiator. A zero value is allowed. In this case there are no Inquiry data transfer and no CHECK CONDITION status.

The Target shall terminate the transfer of the Inquiry Bytes when all bytes requested by the command have been transferred or all available Inquiry Data bytes have been transferred.

If an INQUIRY command is received from an Initiator with a pending unit Attention condition (before the Target reports CHECK CONDITION status), the Target shall perform the INQUIRY command and shall not clear the unit Attention condition.

CHECK CONDITION status shall be reported only when the Target cannot return the Inquiry Data.

The Inquiry Data contains a five-byte Header, followed by PRIVATE USE parameters if any.

INQUIRY DATA

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Device Type							
2nd	RM-Bit	Device Type Qualifier						
3rd	Standard Identification							
4th	0	0	0	0	0	0	0	0
5th	Additional Length							
6th	PRIVATE USE Parameter							
	PRIVATE USE Parameters							
n th	PRIVATE USE Parameters							

The type of devices is identified by the 1st byte as follows:

Direct Access Device	(00)
Sequential Access Device	(01)
Printer Device	(02)
Processor Device	(03)
Write-Once-Multiple-Read Device	(04)
Read-Only-Direct-Access Device	(05)
Reserved for future standardization	(06)-(7E)
Logical Unit not present	(7F)
Private Use	(80)-(FF)

Bit 8 of the second byte of the INQUIRY Data is the Removable Medium Bit (RM-Bit), it indicates:

- if set to ZERO: the medium is not removable
- if set to ONE : the medium is removable.

Bits 7 to 1 of the second byte of the INQUIRY Data, the Device Type Qualifier field, is a seven-bit user-specified identifier generated within the Target. These bits shall be set to ZERO if this facility is not used. When used it allows each user to assign unique identifiers to each peripheral device type that is uspported on the system. The identifier may be used by self-configuring software to know the device type attached to the addressed Logical Unit.

The 3rd byte of the Inquiry Data identifies which version of which standard is implemented by the system.

Bits 8, 7	are reserved for future standardization and shall be set to ZERO.
Bits 6 to 4	shall indicate the version of this ECMA Standard.
Bits 3 to 1	shall indicate the version of the ANSI standard for SCSI.

NOTE 3

In implementation of this edition of this ECMA Standard Bits 6 to 4 shall be set to ZERO ZERO ONE.

6.2.5 RECEIVE DIAGNOSTIC RESULTS Command (1C)

Common command.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	1	0	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	Allocation Length (MSB)							
5th	Allocation Length (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The RECEIVE DIAGNOSTIC RESULTS command requests analysis data be sent to the Initiator after completion of a SEND DIAGNOSTIC command.

The Allocation Length field specifies the number of bytes that the Initiator has allocated for returned diagnostic data. If the value specified by the Allocation Length field equals zero, no diagnostic data shall be transferred. Any other value shall indicate the maximum number of bytes that shall be transferred. The Target shall terminate the DATA IN phase when the number of bytes specified by the Allocation Length field have been transferred or when all available diagnostic data has been transferred to the Initiator.

The contents of diagnostic data returned are not specified by this Standard.

6.2.6 REQUEST SENSE Command (03)

Common command.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	0	1	1
2nd		LUN			0	0	0	0	0
3rd		0	0	0	0	0	0	0	0
4th		0	0	0	0	0	0	0	0
5th		Allocation Length							
6th		Private Use		0	0	0	0	Flag	Link

The REQUEST SENSE command requests that the Target transfers Sense Data to the Initiator.

The Allocation Length field specifies the number of bytes that the Initiator has allocated for the returned Sense Data.

If the value specified by the Allocation Length field equals zero, the number of bytes to be transferred shall be four bytes. For all other values the Target shall terminate when the allocated number of bytes has been transferred or when all available Sense Data has been transferred.

The Target shall preserve Sense Data for the CHECK CONDITION status for the Initiator until the Initiator issues a command again.

If the Target is not capable of preserving the CHECK CONDITION status for each LUN it shall reject any other command with the BUSY COMPLETION status until the Initiator that has received the CHECK CONDITION status issues a command for the same LUN.

The REQUEST SENSE command shall be terminated by the CHECK CONDITION status only if a fatal error occurs in the command execution.

NOTE 4

Examples of fatal errors.

i) The Target receives a byte of a CDB in which a bit which should be set to ZERO by definition is set to ONE.

ii) An unrecovered parity error occurs on the DATA BUS.

iii) A Target malfunction prevents return of the Sense Data.

If any non-fatal error occurs during the execution of the REQUEST SENSE command, the Target shall return the Sense Data with GOOD status.

Following a fatal error on a REQUEST SENSE command, the Sense Data may be invalid.

6.2.6.1 Non-Extended Sense Bytes Format

The format of the Sense Bytes is specified by the first byte (Error Class and Error Type).

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		AV-Bit	Error Class 0-6			Error Type			
2nd		Private Use				Logical Block Address (MSB)			
3rd		Logical Block Address							
4th		Logical Block Address (LSB)							

Bit 8 of the 1st byte, the Address Valid Bit (AV-Bit), if set to ONE indicates that the Logical Block Address field contains valid information related to the specific error.

If set to ZERO, this AV-Bit indicates that no valid information is contained in the Logical Block Address field.

This version of the Standard does not specify the meaning of bits 7-1 of the 1st byte which is left to Private Use.

6.2.6.2 Extended Sense Byte Format

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		V-Bit	1	1	1	Error Type			
2nd		Segment Number							
3rd		FM-Bit	EOM-Bit	ILI-Bit	0	Sense Key			
4th		Information Bytes (MSB)							
5th		Information Bytes							
6th		Information Bytes							
7th		Information Bytes (LSB)							
8th		Additional Sense Length							
n th		Additional Sense Bytes							

Bit 8 of the 1st byte, the Validity Bit (V-Bit), when set to ONE, indicates that the Information Bytes field contain valid information.

When set to ZERO, the Validity Bit indicates that the Information Bytes field contain no defined information.

Bits 7 to 5 of the 1st byte specify Error Class 7.

Bits 4 to 1, the Error Type field, specify:

- (00) specifies the Extended Sense Bytes format,
- (01) to (14) are reserved for future standardization,
- (15) is for Private Use.

The 2nd byte, the Segment Number field, specifies the current segment number of the Extended Sense Bytes in response to a COPY command. Up to 256 segments are supported beginning with segment zero.

Bit 8 of the 3rd byte, the File Mark Bit (FM-Bit), when set to ONE, indicates that the current command reads a File Mark. This bit is used only for sequential access devices.

When set to ZERO, the File Mark Bit shall be ignored.

Bit 7 of the 3rd byte, the End of Medium Bit (EOM-Bit) indicates that the physical end of a medium (i.e. end of tape, beginning of tape, out of paper, etc.) has occurred.

When set to ZERO, the End of Medium Bit shall be ignored. For direct access devices, the End of Medium Bit shall always be set to ZERO (see ILLEGAL REQUEST Sense Key).

Bit 6 of the 3rd byte, the Incorrect Length Indicator Bit (ILI-Bit), when set to ONE, indicates that the requested logical block length is not equal to the logical data length on the medium.

If set to ZERO, the Incorrect Length Indicator Bit shall be ignored.

Bits 4 to 1 of the 3rd byte identify the Sense Key (see 6.2.6.3).

Bytes 4 to 7, the Information Bytes field, specify:

- For direct access devices, write-once-multiple-read devices and read-only-direct-access devices the Information Bytes field specify the unsigned logical block address associated with the Sense Key.
- For sequential access devices, printer devices and processor devices the Information Bytes field normally specifies the difference between the requested length and the actual length in either bytes or blocks, as determined by the command (negative values are indicated by two's complement notation).
- The difference between the number of requested blocks and the number of blocks already copied or compared for the current Segment Descriptor of a COPY, COMPARE or COPY AND VERIFY command.

Byte 8, the Additional Sense Length field, specifies the number of Additional Sense Bytes to follow. The Target shall terminate the transmission of Sense Bytes when the Allocation Length field (defined in the REQUEST SENSE command) has the value zero or when all Sense Bytes have been transferred.

The Additional Sense Bytes contains command-specific and/or peripheral-device-specific data that further define the nature of the CHECK CONDITION status. The COPY command and the SEARCH DATA commands define a standard meaning for some of these bytes.

6.2.6.3 Sense Key Coding

Sense Key	Coded Representation
NO SENSE	(0)
RECOVERED ERROR	(1)
NOT READY	(2)
MEDIUM ERROR	(3)
HARDWARE ERROR	(4)
ILLEGAL REQUEST	(5)
UNIT ATTENTION	(6)
DATA PROTECT	(7)
BLANK CHECK	(8)
PRIVATE USE	(9)
COPY ABORTED	(A)
ABORTED COMMAND	(B)
EQUAL	(C)
VOLUME OVERFLOW	(D)
MISCOMPARE	(E)
Reserved for future standardization	(F)

NO SENSE

This Sense Key indicates that there is no information to be reported for the designated unit. This would be the case for a successful operation or an operation checked because either of FM, EOM or ILI bits being set to ONE.

RECOVERED ERROR

This Sense Key indicates that the last command was completed successfully with some recovery action performed by the Target. Details can be determined by examining the Additional Sense Bytes and the Information Bytes field.

NOT READY

This Sense Key indicates that the addressed Logical Unit cannot be accessed. Operator intervention may be required to correct this condition.

MEDIUM ERROR

This Sense Key indicates that the Target has detected a non-recoverable error on the medium.

HARDWARE ERROR

This Sense Key indicates that the Target has detected a non-recoverable hardware error during the execution of the current command or during a self-test.

ILLEGAL REQUEST

This Sense Key indicates that there was an illegal parameter in the command descriptor block or in the additional parameters supplied for some commands (FORMAT UNIT, SEARCH DATA, etc.). If the Target detects an invalid parameter in the CDB, then it shall terminate the command without altering the medium. The Target may have already altered the medium when it detects an invalid parameter in the data.

UNIT ATTENTION

This Sense Key indicates that the removable medium may have been changed or the Target has been reset (by the BUS DEVICE RESET message or the Hard Reset condition since the last command was issued to the Logical Unit.

This Sense Key shall be reported to each Bus Device the first time that any command is issued after the condition is detected and the requested command will not be executed.

This condition is cleared by the next command (except REQUEST SENSE) from the same Initiator.

DATA PROTECT

This Sense Key indicates that a command which reads or writes the medium was attempted on a block that was protected for this operation. The read or write command was not performed.

BLANK CHECK

This Sense Key indicates that a read command encountered a blank block or a write command encountered a non-blank block (on a write-once-multiple-read device).

PRIVATE USE

This Sense Key is for Private Use.

COPY ABORTED

This Sense Key indicates that a COPY command has been aborted due to an error condition on the Source and/or Destination Bus Device.

ABORTED COMMAND

This Sense Key indicates that the Target has aborted the command. The Initiator may be able to recover by trying the command again.

EQUAL

This Sense Key indicates that a SEARCH DATA EQUAL, SEARCH DATA HIGH or SEARCH DATA LOW command is satisfied.

VOLUME OVERFLOW

This Sense Key indicates that a buffered device has reached the end of the medium and data remains in the buffer which was not written onto the medium. The command RECOVER BUFFERED DATA may be issued to read the unwritten data from the buffer.

MISCOMPARE

This Sense Key indicates that the source data is not identical with the data read from the medium.

6.2.7 SEND DIAGNOSTIC (1D)

Common command.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	1	1	0	1
2nd		LUN			0	0	ST-Bit	DOL-Bit	UOL-Bit
3rd		0	0	0	0	0	0	0	0
4th		Parameter List Length (MSB)							
5th		Parameter List Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

This command requests the Target to perform diagnostic tests on itself and/or attached peripheral devices.

The Parameter List Length field specifies the number of parameter bytes transferred during the DATA OUT phase. The parameter bytes specify Private Use parameters. The value specified by the Parameter List Length field may be zero, in this case no data shall be transferred. This shall not be considered as an error condition.

Bit 1 of the 2nd byte, the Logical Unit Off-Line Bit (UOL-Bit), if set to ONE, enables write operations on the user medium or operations that affect user-visible medium positioning.

Bit 2 of the 2nd byte, the Bus Device Off-Line Bit (DOL-Bit), if set to ONE, enables diagnostic operations which may adversely affect operations to other Logical Units on the same Target.

The Logical Unit Off-Line and Bus Device Off-Line Bits are generally set by operating system software, while the Parameter List Length field is prepared by diagnostic application software. Thus by preventing operations which are not enabled by these bits, the Target assists the operating system in protecting its resources.

Bit 3 of the 2nd byte, the Self Test Bit (ST-Bit), if set to ONE, directs the Target to complete its default self-test. If the self-test is requested, the value specified by the Parameter List Length field shall have the value zero. A RECEIVE DIAGNOSTIC RESULTS command is not required, since if the SEND DIAGNOSTIC command terminates with GOOD STATUS, the self-test shall be considered successfully passed.

6.2.8 TEST UNIT READY (00)

Common command.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

This command provides a means to check whether the addressed Logical Unit is ready. This is not a request for a self-test.

If the Logical Unit will accept a medium access command without returning a CHECK CONDITION status, this command shall return a GOOD status.

6.3 Specific Commands for Direct Access Devices

Name		Coded Representation
FORMAT UNIT	B	(04)
MODE SELECT	Nm	(15)
MODE SENSE	Nm	(1A)
PREVENT/ALLOW MEDIUM REMOVAL	Nm	(1E)
READ	B	(08)
READ EXTENDED	E	(28)
READ CAPACITY	E	(25)
REASSIGN BLOCKS	Nm	(07)
RELEASE	Nm	(17)
RESERVE	Nm	(16)
REZERO UNIT	Nm	(01)
SEARCH DATA HIGH	Nm	(30)
SEARCH DATA EQUAL	Nm	(31)
SEARCH DATA LOW	Nm	(32)
SEEK	Nm	(0B)
SEEK EXTENDED	Nm	(2B)
SET LIMITS	Nm	(33)
START/STOP UNIT	Nm	(1B)
VERIFY	Nm	(2F)
WRITE	B	(0A)
WRITE EXTENDED	E	(2A)
WRITE AND VERIFY	Nm	(2E)

B = mandatory in Basic Level

E = mandatory in Extended Level

Nm= not mandatory in either level.

6.3.1 FORMAT UNIT Command (04)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	1	0	0
2nd	LUN			FD-Bit	CL-Bit	Defect List Format		
3rd	Private Use							
4th	Interleave (MSB)							
5th	Interleave (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The FORMAT UNIT command ensures that the medium is formatted so that all data blocks can be accessed. There is no guarantee that the medium has or has not been altered. In addition, the medium may be certified and control structures may be created for the management of the medium and its defects.

- Bit 5 of the 2nd byte, the Format Data Bit (FD-Bit), when set to ONE, indicates that Format Data is supplied during the DATA OUT phase. The Defect List included with this data specifies the defects which are to be entered into the defect map. The format of the Defect List is determined by the Defect List Format field.

  The Format Data Bit when set to ZERO, indicates that the DATA OUT phase shall not occur, and that no defect data is to be supplied by the Initiator.

- Bit 4 of the 2nd byte, the Complete List Bit (CL-Bit), when set to ONE, indicates that the data supplied by the Initiator is the complete list of known defects. Any previous defect map or defect data shall be erased. The Target may add to this list as it formats the medium.

  The Complete List Bit, when set to ZERO, indicates that the data supplied is in addition to existing defect data using the current format. When using the Block Format, the Defect List refers to the current block size (and not to the new block size, if it is different) and the Defect List refers to current logical block addresses (not physical addresses). The Target may add to this list as it formats the medium.

The Defect List Format field specifies additional information related to the Defect List (see the table below).

The Interleave field requests that the logical blocks be related in a specific fashion to the physical blocks to facilitate speed matching. If the value specified by the Interleave field equals zero, the Target is requested to use its default interleave. If the value specified by the Interleave field equals one consecutive logical blocks shall be placed in consecutive physical order. Values of two or greater are left for Private Use (see 3).

6.3.1.1 Types of FORMAT UNIT Commands

There are several types of FORMAT UNIT commands. Each type is specified by the bit pattern defined by bits 5 to 1 of the 2nd byte of the CDB.

FD-BIT	CL-BIT	DEFECT LIST			
0	X	X	X	X	Format with no defect Data sent from the Initiator to the Target.
1	0	0	X	X	Format adding defects specified in Defect List to known defects. See Defect List Table-Block Format.
1	1	0	X	X	Format using defects in Defect List as full set known defects. See Defect List Table-Block Format.
1	0	1	0	0	Format adding defects in Defect List to known defects. See Defect List Table-Byte From Index Format.
1	1	1	0	0	Format using defects in Defect List as full set known defects. See Defect List Table-Bytes From Index Fromat.
1	0	1	0	1	Format adding defects in Defect List to known defects. See Defect List Table-Physical Sector Format.
1	1	1	0	1	Format using defects in Defect List as full set of known defects. See Defect List Table-Physical Sector F.
1	X	1	1	0	Format not defined.
1	0	1	1	1	Reserved for future standardi-zation.
1	1	1	1	1	Reserved for future standardi-zation.

X = Bit set either to ZERO or to ONE

6.3.1.2 Defect Lists

There are three types of Defect Lists:

- Defect List - Block Format
- Defect List - Byte from Index Format
- Defect List - Physical Sector Format.

These Defect Lists are described below.

6.3.1.2.1 Defect List - Block Format

The Defect List - Block Format consists of a 4-byte Header followed by one or more 4-byte Defect Descriptors.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	0	0	0	0	0	0	0	0
3rd	Defect List Length (MSB)							
4th	Defect List Length (LSB)							

Defect Descriptor

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Defect Block Address (MSB)							
2nd	Defect Block Address							
3rd	Defect Block Address							
4th	Defect Block Address (LSB)							

The Defect List Length field of the header specifies the length in bytes of the Defect Descriptor(s) that follows. The number specified by the Defect List Length field is therefore equal to four times the number of the Defect Descriptors.

The Defect Descriptor for the Block Format consists of a 4-byte Defect Block Address field that specifies the block address of the block containing the defect.

The Defect Descriptors shall be in ascending order of the addresses of the blocks containing a defect.

6.3.1.2.2 Defect List - Bytes from Index Format

The Defect List - Bytes from Index Format consists of a 4-byte Header followed by one or more 8-byte Defect Descriptors.

Header

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	0	0	0	0	0	0	0	0
3rd	Defect List Length (MSB)							
4th	Defect List Length (LSB)							

Defect Descriptor

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	Cylinder Number of Defect (MSB)							
2nd	Cylinder Number of Defect							
3rd	Cylinder Number of Defect (LSB)							
4th	Head Number of Defect							
5th	Defect Byte Location from Index (MSB)							
6th	Defect Byte Location from Index							
7th	Defect Byte Location from Index							
8th	Defect Byte Location from Index (LSB)							

The Defect List Length field of the Header specifies the length in bytes of the Defect Descriptor(s) that follows. The number specified by the Defect List Length is therefore equal to eight times the number of Defect Descriptors.

The Defect Descriptor specifies a defect location comprising the following fields: Cylinder Number of Defect, Head Number of Defect, and the Defect Bytes From Index (i.e. the number of non-defective bytes between the Index and the first defective byte).

The Defect Descriptors shall be sent in ascending order of the defect locations.

In specifying the location of the defect, the Cylinder Number of Defect field is considered most-significant and the Defect Bytes From Index field is considered least-significant.

A Defect Bytes From Index field of (FFFFFFFF) indicates that the entire track shall be re-assigned.

6.3.1.2.3 Defect List - Physical Sector Format

The Defect List - Physical Sector Format consists of a 4-byte header followed by one or more 8-byte Defect Descriptors.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	0	0	0	0	0	0	0	0
3rd	Defect List Length (MSB)							
4th	Defect List Length (LSB)							

Defect Descriptor

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Cylinder Number of Defect (MSB)							
2nd	Cylinder Number of Defect							
3rd	Cylinder Number of Defect (LSB)							
4th	Head Number of Defect							
5th	Defect Sector Number (MSB)							
6th	Defect Sector Number							
7th	Defect Sector Number							
8th	Defect Sector Number (LSB)							

The Defect List Length field of the header specifies the length in bytes of the Defect Descriptors that follow. The number specified by the Defect List Length is therefore equal to eight times the number of Defect Descriptors.

The Defect Descriptor specifies a defect location comprising the following fields: Cylinder Number of Defect, the Head Number of Defect, and the Defect Sector Number.

The Defect Descriptors shall be sent in ascending order of the defect locations.

In specifying the location of the defect, the Cylinder Number of Defect field is considered most-significant and the Defect Sector Number field is considered least-significant.

A Defect Sector Number field of (FFFFFFFF) indicates that the entire track shall be re-assigned.

6.3.2 MODE SELECT Command (15)

Specific command for direct access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	1	0	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Parameter List Length							
6th	Private Use		0	0	0	0	Flag	Link

The MODE SELECT command provides a means to specify parameters of the medium, of the Logical Unit, or of peripheral devices.

The Parameter List Length field specifies the length in bytes of the Parameter List which shall be transferred during the DATA OUT phase. If the value specified by the Parameter List Length field equals zero, no data shall be transferred. This shall not be considered an error condition.

The Parameter List contains a 4-byte header, followed by zero or more 8-byte Block Descriptors, followed by zero or more Private Use parameters, if any.

Header

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	Medium Type							
3rd	0	0	0	0	0	0	0	0
4th	Block Descriptor Length							

Block Descriptor

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		Density Indicator							
2nd		Number of Blocks (MSB)							
3rd		Number of Blocks							
4th		Number of Blocks (LSB)							
5th		0	0	0	0	0	0	0	0
6th		Block Length (MSB)							
7th		Block Length							
8th		Block Length (LSB)							

Private Use parameter(s)

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		Private Use Parameter							
n th		Private Use Parameter							

Byte 2 of the Header identifies the Medium Types as follows.

If set to (00) : Default medium type

If set to (01) to (08) : Not specified by this Standard

If set to (09) : ECMA-54 - 200 mm Flexible Disk Cartridge using Two-Frequency Recording at 13262 ftprad on One Side

If set to (0A) ECMA-59 - 200 mm Flexible Disk Cartridges using Two-Frequency Recording at 13626 ftprad on Both Sides

If set to (0B) : ECMA-69 - 200 mm Flexible Disk Cartridges using MFM Recording at 13262 ftprad on Both Sides

If set to (0C) to (0D) : Not specified by this Standard

If set to (0E) : ECMA-66 - 130 mm Flexible Disk Cartridges using Two-Frequency Recording at 7958 ftprad on One Side

If set to (0F) to (11) : Not specified by this Standard

If set to (12) : ECMA-70 - 130 mm Flexible Disk Cartridges using MFM Recording at 7958 ftprad on Both Sides; 1,9 Tracks per mm

If set to (13) to (15) : Not specified by this Standard

If set to (16) : ECMA-78 - 130 mm Flexible Disk Cartridges using MFM Recording at 7958 ftprad on Both Sides; 3,8 Tracks per mm

If set to (17) to (19) : Not specified by this Standard

If set to (1A) : ECMA-99 - 130 mm Flexible Disk Cartridges using MFM Recording at 13262 ftprad on Both Sides; 3,8 Tracks per mm

If set to (1B) to (1D) : Not specified by this Standard

If set to (1E) : ECMA-100 - 90 mm Flexible Disk Cartridges using MFM Recording at 7958 ftprad on Both Sides; 5,3 Tracks per mm

Reserved for future standardization : (1F) to (7F)

Private Use : (80) to (FF)

Byte 4 of the header, the Block Descriptor Length field, specifies the total length in bytes of all the Block Descriptors. The number specified by the Block Descriptor Length field is therefore equal to eight times the number of Block Descriptors. If the value specified by the Block Descriptor Length field equals zero, no Block Descriptors are included in the Parameter List. This shall not be considered an error condition.

The Block Descriptor specifies the medium characteristics for all or part of a Logical Unit. Each Block Descriptor contains a Number of Blocks field and a Block Length field.

The Density Indicator field specifies the physical recording density as follows.

If set to (00) : Default density of the medium

If set to (01) : 7958 flux transitions per radian

If set to (02) : 13262 flux transitions per radian

If set to (03) : 15916 flux transitions per radian

The values (04) to (7F) are reserved for future standardization

The values (80) to (FF) are for Private Use.

The Number of Blocks field specifies the length in logical blocks of the medium that corresponds to the specified Density Identification and has blocks of a length equal to that specified in the Block Length field of the Block Descriptor. If the value specified by the Number of Blocks field equals zero, this shall indicate all the remaining logical blocks of the Logical Unit.

The Block Length field specifies the length in bytes of the logical block.

6.3.3 MODE SENSE Command (1A)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	0	1	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Allocation Length							
6th	Private Use		0	0	0	0	Flag	Link

The MODE SENSE command provides a means for a Target to report the parameters of its medium, Logical Unit, or peripheral device. It is a complementary command to the MODE SELECT command for support of a medium that may contain multiple block sizes or densities.

The Allocation Length field specifies the number of bytes that the Initiator has allocated for returned Sense Data. If the value specified by the Allocation Length field equals zero, this indicates that no Sense Data shall be transferred. This shall not be considered an error condition. Any other value shall indicate the number of bytes that shall be transferred. The Target shall terminate the DATA IN phase when a number of bytes equal to the number specified by the Allocation Length field has been transferred or when all available Sense Data has been transferred to the Initiator.

The Mode Sense Data comprises a 4-byte Header, followed by zero or more 8-byte Block Descriptors, followed by Private Use parameters, if any.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Sense Data Length							
2nd	Medium Type							
3rd	WP-Bit	0	0	0	0	0	0	0
4th	Block Descriptor Length							

Block Descriptor

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		Density Indicator							
2nd		Number of Blocks (MSB)							
3rd		Number of Blocks							
4th		Number of Blocks (LSB)							
5th		0	0	0	0	0	0	0	0
6th		Block Length (MSB)							
7th		Block Length							
8th		Block Length (LSB)							

Private Use parameter(s)

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		Private Use Parameter							
n th		Private Use Parameter							

Byte 1 of the Header, the Sense Data Length field, specifies the length in bytes of the Mode Sense Data which is available for transfer during the DATA IN phase.

Byte 2 of the Header shall identify the Medium Types as follows.

If set to (00) : Default medium type

If set to (01) to (08): Not specified by this Standard

If set to (09) : ECMA-54 - 200 mm Flexible Disk Cartridge using Two-Frequency Recording at 13262 ftprad on One Side

If set to (0A) : ECMA-59 - 200 mm Flexible Disk Cartridges using Two-Frequency Recording at 13262 ftprad on Both Sides

If set to (0B) : ECMA-69 - 200 mm Flexible Disk Cartridges using MFM Recording at 13262 ftprad on Both Sides

If set to (0C) to (0D): Not specified by this Standard

If set to (0E) : ECMA-66 - 130 mm Flexuble Disk Cartridges using Two-Frequency Recording at 7958 ftprad on One Side

If set to (0F) to (11): Not specified by this Standard

If set to (12) : ECMA-70 - 130 mm Flexible Disk Cartridges using MFM Recording at 7958 ftprad on Both Sides; 1,9 Tracks per mm

If set to (13) to (15): Not specified by this Standard

If set to /16) : ECMA-78 - 130 mm Flexible Disk Cartridges using MFM Recording at 7958 ftprad on Both Sides; 3,8 Tracks per mm

If set to (17) to (19): Not specified by this Standard

If set to (1A) : ECMA-99 - 130 mm Flexible Disk Cartridges using MFM Recording at 13262 ftprad on Both Sides; 3,8 Tracks per mm

If set to (1B) to (1D): Not specified by this Standard

If set to (1E) : ECMA-100 - 90 mm Flexible Disk Cartridges using MFM Recording at 7859 ftprad on Both Sides; 5,3 Tracks per mm

Reserved for future standardization : (1F) to (7F)

Private Use : (80) to (FF)

If bit 8 of the 3rd byte of the Header, the Write-Protect Bit (WP Bit) is set to ZERO, writing on the medium is possible. If it is set to ONE writing on the medium is inhibited.

Byte 4 of the Header, the Block Descriptor Length field, specifies the total length in bytes of all the Block Descriptors. The number specified by the Block Descriptor Length field is therefore equal to eight times the number of Block Descriptors. If the value specified by the Block Descriptor Length field equals zero, no Block Descriptors are included in the Parameter List. This shall not be considered an error condition.

The Block Descriptor specifies the medium characteristics for all or part of a Logical Unit. Each Block Descriptor contains a Number of Blocks field and a Block Length field.

The Density Indicator field specifies the physical recording density as follows.

If set to (00) : Default density of the medium

If set to (01) : 7958 flux transitions per radian

If set to (02) : 13262 flux transitions per radian

If set to (03) : 15916 flux transitions per radian

The values (04) to (7F) are reserved for future standardization.
The values (80) to (FF) are for Pirvate Use.

The Number of Blocks field specifies the number of logical blocks of the medium that corresponds to the specified Density Identification and has blocks of a length equal to that specified in the Block Length field of the Block Descriptor. If the value specified by the Number of Blocks field equals zero, it indicates all the remaining logical blocks of the Logical Unit.

The Block Length field specifies the length in bytes of the logical block.

6.3.4 PREVENT/ALLOW MEDIUM REMOVAL Command (1E)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	1	1	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	Prevent Bit
6th	Private Use		0	0	0	0	Flag	Link

The PREVENT/ALLOW MEDIUM REMOVAL command requests that a Target enables or disables the removal of the medium in the Logical Unit.

Bit 1 of the 5th byte, the Prevent Bit, if set to ONE, indicates that mechanisms which normally allow removal of the medium are inhibited. The Prevent Bit, if set to ZERO, indicates that these mechanisms are not inhibited, so that the medium can be removed.

If the Prevent Bit has been set to ONE, the medium cannot be removed until receipt of a PREVENT/ALLOW MEDIUM REMOVAL command with the Prevent Bit set to ZERO, or by a RESET BUS DEVICE message from any Initiator or by a Hard Reset condition.

6.3.5 READ Command (08)

Specific command for direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	1	0	0	0
2nd		L U N			Logical Block Address (MSB)				
3rd		Logical Block Address							
4th		Logical Block Address (LSB)							
5th		Transfer Length							
6th		Private Use		0	0	0	0	Flag	Link

The READ command requests that the Target transfer data to the Initiator.

The Logical Block Address field specifies the address of the logical block where the read operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value specified by the Transfer Length field equals zero, this means that 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks to be transferred.

The most recent data value written shall be returned.

6.3.6 READ EXTENDED Command (28)

Specific command for direct access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	0	0	0
2nd	LUN			0	0	0	0	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The READ EXTENDED command requests that the Target transfer data to the Initiator.

Bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit), is set to ONE to indicate that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

The Logical Block Address field specifies the address of the logical block where the read operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value specified by the Transfer Length field equals zero, no logical blocks shall be transferred. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be transferred.

The most recent data value written shall be returned.

6.3.7 READ CAPACITY Command (25)

Specific command for direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	1	0	0	1	0	1
2nd		L U N			0	0	0	0	RA-Bit
3rd		Logical Block Address (MSB)							
4th		Logical Block Address							
5th		Logical Block Address							
6th		Logical Block Address (LSB)							
7th		0	0	0	0	0	0	0	0
8th		0	0	0	0	0	0	0	0
9th		Private Use		0	0	0	0	0	PMI
10th		Private Use		0	0	0	0	Flag	Link

The READ CAPACITY command provides a means for the Initiator to request information regarding the capacity of the Logical Unit.

Bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit), is set to ONE to indicate that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

Bit 1 of the 9th byte, the Partial Medium Indicator Bit (PMI), when set to ZERO indicates that the information returned in the Read Capacity Data shall be the logical block address and block length (in bytes) of the last logical block of the Logical Unit. The value specified by the Logical Block Address field shall be zero in this case.

When the PMI Bit is set to ONE, this indicates that the information returned shall be the logical block address and block length (in bytes) of the last logical block after the logical block the address of which is specified by the Logical Block Address field in the CDB before a substantial delay in data transfer will be encountered (e.g. a cylinder boundary).

The following eight bytes of Read Capacity Data shall be sent by the Target during the DATA IN phase of the command:

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Logical Block Address (MSB)							
2nd	Logical Block Address							
3rd	Logical Block Address							
4th	Logical Block Address (LSB)							
5th	Block Length (MSB)							
6th	Block Length							
7th	Block Length							
8th	Block Length (LSB)							

6.3.8 REASSIGN BLOCKS Command (07)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	1	1	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

The REASSIGN BLOCKS command requests the Target to reassign the defective logical blocks to an area on the Logical Unit reserved for this purpose.

The Initiator transfers a Defect List which contains the logical block addresses of the blocks to be reassigned. The Target shall reassign the physical medium used for each logical block address in the list. The data in the logical blocks the addresses of which are specified in the Defect List may be altered, but the data in all other logical blocks on the medium shall be preserved.

The Defect List contains a 4-byte Header followed by one or more 4-byte Defect Descriptors.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	0	0	0	0	0	0	0	0
3rd	Defect List Length (MSB)							
4th	Defect List Length (LSB)							

Defect Descriptor

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Defect Logical Block Address (MSB)							
2nd	Defect Logical Block Address							
3rd	Defect Logical Block Address							
4th	Defect Logical Block Address (LSB)							

The Defect List Length field specifies the length in bytes of the Defect Descriptors that follow. The number specified by the Defect List Length field is equal to four times the number of Defect Descriptors.

The Defect Descriptor for the logical block address specifies a 4-byte Defect Logical Block Address field that specifies the block containing the defect. The Defect Descriptor shall be in ascending order of defect locations.

If the Logical Unit has insufficient capacity to reassign all of the defective logical blocks, the command shall terminate with a CHECK CONDITION status and the Sense Key set to MEDIUM ERROR. The logical block address of the last logical block reassigned shall be returned in the Sense Data and the Validity bit in the Extended Sense Bytes shall be set to ONE.

6.3.9 RELEASE Command (17)

Specific command for direct access devices.

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	1	1
2nd		L U N			3rdP-Bit	3rd Party Device ID			Ex-Bit
3rd		Reservation Identification							
4th		0	0	0	0	0	0	0	0
5th		0	0	0	0	0	0	0	0
6th		Private Use		0	0	0	0	Flag	Link

The RELEASE command is used to release previously reserved Logical Units, or, if the Extent Release option is implemented, previously reserved extents within Logical Units. It is not an error for an Initiator to attempt to release a reservation which is not currently active or queued.

6.3.9.1 Logical Unit Release

Bit 1 of the 1st byte, the Extent Bit (Ex-Bit), when set to ZERO, indicates that this command shall cause the Target to terminate any active reservation from the Initiator to the specified Logical Unit.

6.3.9.2 Extent Release

If the Extent Bit is set to ONE and Extent Reservation is not implemented (see 6.3.10, RESERVE command), then the RELEASE command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

If the Extent Bit is set to ONE and the Extent Reservation option is implemented, this command shall cause any reservation from the requesting Initiator with a matching Reservation Identification which is active or queued to be terminated. Other reservations from the requesting Initiator shall remain in effect.

If the Logical Unit queues reservations, then when a RELEASE command is processed, the reservation queue shall be examined on a first-in first-out basis. If there are one or more reservations in the queue which can now be activated, the Logical Unit shall first disconnect from the Initiator. It shall then successively reconnect with each Initiator the queued reservation of which may now be activated. A queued reservations request shall not be activated if it conflicts with any previously queued reservation. After first granting all

possible queued reservations, the unit shall reconnect with the Initiator of the RELEASE command.

If a Logical Unit which queues reservations receives a RELEASE command from a second Initiator while it is disconnected during processing of a previous RELEASE command, it shall then disconnect from the second Initiator and suspend processing of the second RELEASE commande until after reconnection with the first Initiator, or until it is determined that reconnection has failed.

6.3.9.3 Third Party Release

The Third Party Release allows an Initiator to release a Logical Unit or extents within a unit that were previously reserved using Third Party Reservation (see 6.3.10, RESERVE command). This is intended for use in multiple-host systems which use the COPY command.

Bit 5 of the 2nd byte, the Third Party Bit (3rdP-Bit), when set to ZERO, indicates that the RELEASE command operates as stated above. If the Third Party Bit is set to ONE and if the Target does not implement the Third Party Release option, then the Target shall terminate the command with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

If the Third Party Bit is set to ONE and if the Target implements the Third Party Release option, then the RELEASE command operates as stated above, except that the reservation(s) released (if any) shall be restricted to those made previously using the Third Party Reservation option by the same Initiator for the same Bus Device as specified by bits 4 to 2 of the 2nd byte, the Third Party Device ID field.

6.3.10 RESERVE Command (16)

Specific command for direct access devices.

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	1	0
2nd		LUN			3rdP-Bit	3rd Party Device ID			Ex-Bit
3rd		Reservation Identification							
4th		Extent List Length (MSB)							
5th		Extent List Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The RESERVE command is used to reserve Logical Units, or, if the Extent Reservation option is implemented, extents within Logical Units for the use of the Initiator.

6.3.10.1 Logical Unit Reservation

When bit 1 of the 2nd byte, the Extent Bit (Ex-Bit), is set to ZERO, this command shall request that the entire Logical Unit be reserved for the exclusive use of the Initiator until the reservation is released by a RELEASE command from the same Initiator, by a RESET BUS DEVICE message from any Inititator, or by a Hard Reset condition. A Logical Unit reservation shall not be granted if any extent or Logical Unit is reserved by another Initiator or if any extent with a read shared reservation type is reserved by this Initiator. It shall be permissible for an Initiator to reserve a Logical Unit that is currently reserved for that Initiator. If the Extent Bit is set to ZERO, the values of the Reservation Identification and the Extent List Length fields shall be ignored.

If the Logical Unit, or any extent within the Logical Unit is reserved for another Initiator, or a read shared extent reservation exists from this Initiator, the Target shall respond by either:

- a RESERVATION CONFLICT status indication, or
- queueing the reservation request and then disconnecting until all previously queued reservations have been released and the Logical Unit is available, then reconnecting to perform the reservation.

When the queueing capacity is reached, subsequent attempts to perform any operation on the reserved Logical Unit, that command shall be rejected with RESERVATION CONFLICT status.

6.3.10.2 Extent Reservation

The 3rd byte, the Reservation Identification field, provides a means for an Initiator to identify each Extent Reservation. This allows an Initiator in a multiple-tasking environment, to have multiple reservations outstanding. The Reservation Identification field is used in the RELEASE command to specify which reservation is to be released. It is valid only if the Extent Bit is set to ONE.

If Extent Reservation is implemented, extents within the Logical Unit may be reserved, each with a separate Reservation Type. If the reservation cannot be granted because of conflicts with a previous reservation, then it is queued until it is allowed to be active. Reservations are only made active when all extents are free from conflict with active or previously queued reservations.

If the Extent Bit is set to ONE, and Extent Reservation is implemented, then:

- The Extent List shall be checked for the number of extents in the reservation request. If the Extent List contains more extents than are supported on the Logical Unit, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST. If the Extent List contains more extents than are currently available on the Logical Unit, then the request shall be queued and the Target shall disconnect until sufficient extents are made available (by means of the RELEASE command). When sufficient extents are available, the Target shall reconnect to continue processing the request. If the value specified by the Extent List Length field equals zero, no reservation shall be changed and this condition shall not be treated as an error.
- The Extent List shall be checked for validity of the logical block addresses within the extent. If any logical block address specified in the Extent List is invalid for this Logical Unit, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST. The Extent List shall be checked for extent overlaps and if overlaps are found, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.
- If there is already an active or queued reservation for the Logical Unit from this Initiator with the same Reservation Identification, the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

- If the requested reservation does not conflict with any active or previously requested reservation, then the extents specified shall be reserved until released by a RELEASE command from this Initiator or by a RESET BUS DEVICE message from any Initiator or a Hard Reset Condition. If either of the last two conditions occurs, the next command from each Initiator shall be terminated with a CHECK CONDITION status and the Sense Key set to UNIT ATTENTION.
- If the reservation request conflicts with a reservation already active or a reservation request which is already queued, then that request shall be queued and the Target shall disconnect until all conflicting reservations are released (by means of the RELEASE command). When the reservation is granted, the Target shall reconnect to report completion status.

If the Extent Bit is set to ONE, and if the Extent Reservation option is not implemented, then the RESERVE command shall be rejected with CHECK CONDITION status and a Sense Key set to ILLEGAL REQUEST.

The format of the Extent Descriptors shall be as follows:

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	RA-Bit	Reservation Type	
2nd		Number of Blocks (MSB)							
3rd		Number of Blocks							
4th		Number of Blocks (LSB)							
5th		Logical Block Address (MSB)							
6th		Logical Block Address							
7th		Logical Block Address							
8th		Logical Block Address (LSB)							

The size of the Extent List shall be defined by the Extent List Length field. The Extent List shall consist of one or more descriptors as shown in the above table. The Extent Descriptor defines an extent beginning at the logical block address specified by the Logical Block Address field for the number of blocks specified by the Number of Blocks field. If the value specified by the Number of Blocks field equals zero, the extent shall begin at the logical block address specified by the Logical Block Address field and continue through the last logical block address on the Logical Unit.

Bits 1, 2 of the 1st byte of an Extent Descriptor, the Reservation Type field, specifies the type of reservation for each extent. Four types of reservations are possible as follows:

Bit 2	Bit 1	Reservation Type
1	0	Read Exclusive
0	1	Write Exclusive
1	1	Read or Write Exclusive
0	0	Read Shared

If an Extent Reservation from one Initiator is already active or queued when Extent Reservation that overlaps this extent is requested by another Initiator, this requested Extent Reservation will conflict with the previous extent reservation according to the table below.

Reservation Conflict Table

Extent Reservation from other Initiator	Extent Reservation from one Initiator			
	Read Exclusive	Write Exclusive	Read or Write Exclusive	Read Shared
Read Exclusive	Conflict		Conflict	Conflict
Write Exlusive		Conflict	Conflict	Conflict
Read or Write Exclusive	Conflict	Conflict	Conflict	Conflict
Read Shared	Conflict	Conflict	Conflict	

If an extent reservation from one Initiator is already active when a read or write operation is requested by another Initiator, the operation will be rejected with RESERVATION CONFLICT according to the table below.

	Extent Reservation from one Initiator			
Operation Requested by other Initiator	Read Exclusive	Write Exclusive	Exclusive Access	Read Shared
Read	Rejected		Rejected	
Write		Rejected		Rejected

Bit 3 of the 1st byte of an Extent Descriptor, the Relative Address Bit (RA-Bit), when set to ONE indicates that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

If an Initiator sends a command addressed to a logical block which has been reserved and access to which is prohibited by the reservation, then the command shall not be performed and the command shall terminate with RESERVATION CONFLICT status. If a reservation conflict preeludes any part of the command, no part of the command shall be performed. COPY commands shall be terminated with a CHECK CONDITION status and the Sense Key set to DATA PROTECT if any part of the COPY operation is prohibited by an active reservation. If any extent in a Logical Unit is reserved in any way, a FORMAT UNIT command shall be rejected with RESERVATION CONFLICT status.

6.3.10.3 Third Party Reservation

The Third Party Reservation allows an Initiator to reserve a unit or extents within a unit for another Bus Device. This is intended for use in multiple-host systems which use the COPY command. Any Target which implements Third Party Reservation shall also implement Third Party Release (see the RELEASE command).

If bit 5 of the 2nd byte of the CDB, the Third Party Bit (3rdP-Bit) is set to ONE and if Third Party Reservation is not implemented, then the Target shall reject the RESERVE command with CHECK CONDITION status and a Sense Key of ILLEGAL REQUEST.

If the Third Party Bit is set to ZERO, then the RESERVE command operates as described above. If Third Party Reservation is implemented and the Third Party Bit is set to ONE, then the RESERVE command operates as described above except that the Logical Unit or extents are not reserved to the requesting Initiator. Instead, they are reserved to the Bus Device specified by the Third Party Device ID field. The Target shall preserve the reservation until it is released by the same Initiator (or by a RESET BUS DEVICE message from any Initiator or a Hard Reset condition). The Target shall ignore any attempt to release the reservation made by any other Initiator.

6.3.11 REZERO UNIT Command (01)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

The REZERO UNIT command requests that the Target sets the Logical Unit to a particular state. This Standard does not specify this state.

6.3.12 SEARCH EQUAL Command (31)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	1	0	0	0	1
2nd	L U N			I-Bit	Record Format		SD-Bit	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The SEARCH DATA EQUAL command is used to search one or more logical blocks for equality to a data pattern.

The concept of records within a logical block is used to allow multiple records within a logical block to be searched.

Bit 5 of the 2nd byte, the Invertion Bit (I-Bit), if set to ONE, indicates that the search condition is to be inverted. Thus, a SEARCH DATA EQUAL command would search for not-equal data; a SEARCH DATA HIGH command would search for less than or equal data; and a SEARCH DATA LOW command would search for greater than or equal data.

Bits 4, 3 of the 2nd byte, the Record Format field, if set both to ZERO, indicates that fixed-length records are to be searched. The Record Format field, if set to another bit combination, indicates that variable-length records are to be searched.

If the value specified by the Record Format field does not equal zero, it specifies the length of the Variable-Length Indicator field which occurs at the beginning of each variable-length record. Values of one, two or three in the Record Format field specify a one, two or four-byte Variable-Length Indicator field, respectively. The value contained in the Variable-Length Indicator field is the record size in bytes (including the Variable-Length Indicator itself).

If bit 2 of the 2nd byte, the Spanned Data Bit (SD-Bit) is set to ZERO, each record must be wholly contained within a single block. Any space at the end of a block which is

smaller than the record size is ignored by the SEARCH DATA commands. If the Spanned Data Bit is set to ONE, it indicates that the records are spanned, i.e. that they extend over more than one block. Thus, a record may start in one block and end in the next or a subsequent block.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) is set to ONE, it indicates that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

If the 8th and 9th bytes, the Transfer Length field, are set to ZERO, no data shall be searched. This condition shall be treated as an unsuccessful search.

The Link Bit, if set to ZERO, indicates a non-linked command and if the search is satisfied, the command shall be terminated with a CONDITION MET status. A REQUEST SENSE command can then be issued to determine the logical block address and record offset of the matching record.

The Link Bit, if set to ONE, indicates that a command is linked to the SEARCH DATA command and if the search is successful, the next command is executed. If the Relative Address Bit in the next command is set to ONE, the value of the Logical Block Address field of the next command is used as a displacement from the logical block address at which the search was satisfied. If a linked search is not satisfied, the command is terminated with a CHECK CONDITION status. A REQUEST SENSE command can then be issued.

A REQUEST SENSE command following a successful SEARCH DATA command shall:

- return a Sense Key set to EQUAL if the search was satisfied by an exact match. If the search was satisfied by an inequality then a Sense Key set to NO SENSE shall be returned,
- set the Validity Bit to ONE,
- return in the Information Bytes the logical block address of the logical block containing the first matching record,
- return the record offset of the matching record in the first four bytes of Additional Sense Bytes.

A REQUEST SENSE command following an unsuccessful SEARCH DATA command shall:

- return a Sense Key set to NO SENSE, if no errors occurred during command execution,
- set the Validity Bit to ZERO.

The Search Data Parameter List contains a 14-byte Header, followed by one or more Search Argument Descriptors, each followed by n bytes of pattern of data.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Logical Record Size (MSB)							
2nd	Logical Record Size							
3rd	Logical Record Size							
4th	Logical Record Size (LSB)							
5th	First Record Offset (MSB)							
6th	First Record Offset							
7th	First Record Offset							
8th	First Record Offset (LSB)							
9th	Number of Records (MSB)							
10th	Number of Records							
11th	Number of Records							
12th	Number of Records (LSB)							
13th	Search Argument Length (MSB)							
14th	Search Argument Length (LSB)							

Search Argument Descriptor(s)

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Displacement (MSB)							
2nd	Displacement							
3rd	Displacement							
4th	Displacement (LSB)							
5th	Pattern Length (MSB)							
6th	Pattern Length (LSB)							

Patterns of Data

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Pattern of Data							
n th	Pattern of Data							

The Logical Record Size field specifies the record size in bytes for fixed-length records or the maximum record size for variable-length records.

The First Record Offset field specifies the number of bytes to be ignored in the first logical block before the search begins. Subsequent logical blocks are searched beginning with the first byte in the logical block. This permits one or more records to be skipped initially.

The Number of Records field specifies the maximum number of records to be searched by this command. An unsatisfied search shall terminate when the number of records or blocks specified in the Number of Records field of the header of the Search Data Parameter List or in the Number of Blocks field of the CDB, respectively, have been exhausted.

The Search Argument Length field specifies the total length in bytes of all the Search Argument Descriptors which follow. Since the value of the Pattern Length field can vary there is no fixed arithmetical relation between the number of Search Argument Descriptors and the value specified in the Search Argument Length field.

Each Search Argument Descriptor specifies one or more search conditions to be satisfied within a single record in order to satisfy the search. The Search Argument Descriptor comprises the Displacement field and the Pattern Length field, and is followed by the Pattern of Data bytes.

The Displacement field specifies the displacement in bytes of the first byte of the data to be compared from the start of the logical record.

The Pattern Length field specifies the number of the Patterns of Data, i.e. the number of bytes, which follow.

Patterns of Data specify the data to be compared to the logical record.

6.3.13 SEARCH DATA HIGH Command (30)

Specific command for direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	1	1	0	0	0	0
2nd		L U N			I-Bit	Record Format		SD-Bit	RA-Bit
3rd		Logical Block Address (MSB)							
4th		Logical Block Address							
5th		Logical Block Address							
6th		Logical Block Address (LSB)							
7th		0	0	0	0	0	0	0	0
8th		Transfer Length (MSB)							
9th		Transfer Length (LSB)							
10th		Private Use		0	0	0	0	Flag	Link

The SEARCH DATA HIGH command performs the same function as the SEARCH DATA EQUAL command, but is satisfied by a comparison of value of the data on the Peripheral Device being higher than the value of the data in the Pattern of Data bytes.

The fields and the individual bits of this command have the meaning specified for the SEARCH DATA EQUAL command in 6.3.12.

Its Search Data Parameter List has the same structure as that of the SEARCH DATA EQUAL command.

6.3.14 SEARCH DATA LOW Command (32)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	1	0	0	1	0
2nd	L U N			I-Bit	Record Format		SD-Bit	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use	0	0	0	0	Flag	Link	

The SEARCH DATA LOW command performs the same function as the SEARCH DATA EQUAL command, but is satisfied by a comparison of the value of the data on the Peripheral Device being lower than the value of the data in the Pattern of Data bytes.

The fields and the individual bits of this command have the meaning specified for the SEARCH DATA EQUAL command in 6.3.12.

Its Search Data Parameter List has the same structure as that of the SEARCH DATA EQUAL command.

6.3.15 SEEK Command (0B)

Specific command for direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	1	0	1	1
2nd		L U N			Logical Block Address (MSB)				
3rd		Logical Block Address							
4th		Logical Block Address (LSB)							
5th		0	0	0	0	0	0	0	0
6th		Private Use			0	0	0	Flag	Link

The SEEK command requests the Logical Unit to go to the logical block the address of which is specified by the Logical Block Address field.

6.3.16 SEEK EXTENDED Command (2B)

Specific command for direct accesss devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	0	1	1
2nd	LUN			0	0	0	0	0
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	0	0	0	0	0	0	0	0
9th	0	0	0	0	0	0	0	0
10th	Private Use		0	0	0	0	Flag	Link

The SEEK EXTENDED command requests the Logical Unit to go to the logical block the address of which is specified by the Logical Block Address field.

6.3.17 SET LIMITS Command (33)

Specific command for direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	1	1	0	0	1	1
2nd		LUN			0	0	0	RI Bit	WI-Bit
3rd		Logical Block Address (MSB)							
4th		Logical Block Address							
5th		Logical Block Address							
6th		Logical Block Address (LSB)							
7th		0	0	0	0	0	0	0	0
8th		Transfer Length (MSB)							
9th		Transfer Length (LSB)							
10th		Private Use		0	0	0	0	Flag	Link

The SET LIMITS command defines the boundary outside which any subsequent linked command shall not operate. A second SET LIMITS command may not be linked to a chain of commands in which a SET LIMITS command has already been issued.

When bit 2 of the 2nd byte, the Read Inhibit Bit (RI-Bit) is set to ONE, read operations within the declared range are inhibited. When bit 1 of the 2nd byte, the Write Inhibit Bit (WI-Bit) is set to ONE, write operations within the declared range are inhibited.

The Logical Block Address field specifies the starting address (lower boundary) for the range.

The Transfer Length field specifies the range in logical blocks over which the subsequent linked commands may operate. If the value specified by the Transfer Length equals zero, the range shall extend to the last logical block on the Logical Unit.

Any attempt to access outside the declared range shall not be performed or any attempt to perform an inhibited operation within the declared range shall not be undertaken and the command shall be terminated with a CHECK CONDITION status.

6.3.18 START/STOP UNIT Command (1B)

Specific command for direct access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	0	1	1
2nd	LUN			0	0	0	0	IMM-Bit
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	Start-Bit
6th	Private Use		0	0	0	0	Flag	Link

The START/STOP UNIT command requests that a Target enable or disable a Logical Unit for further operations.

If bit 1 of the 2nd byte, the Immediate Bit (I-Bit) is set to ONE, the command is terminated as soon as the sequence is initiated. If the Immediate Bit is set to ZERO, the command is terminated when the Logical Unit has completed the sequence.

Bit 1 of the 5th byte, the Start Bit, if set to ONE, requests the Logical Unit be made ready for use. The Start Bit, if set to ZERO, requests that the Logical Unit be stopped.

6.3.19 VERIFY Command (2F)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	1	1	1
2nd	L U N			0	0	0	BC-Bit	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Verification Length (MSB)							
9th	Verification Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The VERIFY command requests that the Target verify the data written on the medium.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) is set to ONE it indicates that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requies that a previous command in the linked group has accessed a block of data on the Logical Unit.

When bit 2 of the 2nd byte, the Byte Check Bit (BC-Bit), is set to ZERO, the verification is simply a medium verification (CRC, ECC, etc.). When this Byte Check Bit is set to ONE, the verification is a byte-by-byte comparison of data on the medium and the data transferred from the Initiator.

The Logical Block Address field specifies the address of the logical block where the verify operation shall begin.

The Verification Length field specifies the number of contiguous logical blocks of data to be verified. If the value specified by the Transfer Length field equals zero, no logical blocks shall be verified. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be verified.

6.3.20 WRITE Command (0A)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	0	1	0
2nd	L U N			Logical Block Address (MSB)				
3rd	Logical Block Address							
4th	Logical Block Address (LSB)							
5th	Transfer Length							
6th	Private Use		0	0	0	0	Flag	Link

The WRITE command requests that the Target write the data transferred by the Initiator on the medium.

The Logical Block Address field specifies the address of the logical block where the write operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value specified by the Transfer Length field equals zero, 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks to be transferred.

6.3.21 WRITE EXTENDED Command (2A)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	0	1	0
2nd	LUN			0	0	0	0	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The WRITE EXTENDED command requests that the Target write the data transferred by the Initiator on the medium.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) is set to ONE, it indicates that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

The Logical Block Address field specifies the address of the logical block where the write operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of the data to be transferred. If the value of the Transfer Length field equals zero, no logical blocks shall be transferred. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be transferred.

6.3.22 WRITE AND VERIFY Command (2E)

Specific command for direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	1	1	0
2nd	L U N			0	0	0	BC-Bit	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The WRITE AND VERIFY command requests that the Target write the data transferred by the Initiator on the medium and then verify the data written.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) field is set to ONE, it indicates that the Logical Block Address fiels is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

When bit 2 of the 2nd byte, the Byte Check Bit (BC-Bit) is set to ZERO, the verification is simply a medium verification (CRC, ECC, etc.). When this Byte Check Bit is set to ONE, the verification is a byte-by-byte comparison of data written on the Peripheral Device and the data transferred from the Initiator.

The Logical Block Address field specifies the address of the logical block where the write operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value of the Transfer Length field equals zero, no logical blocks shall be transferred. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be transferred.

6.4 Specific Commands for Sequential Access Devices

Name		Coded Representation
ERASE	Nm	(19)
LOAD/UNLOAD	Nm	(1B)
MODE SELECT	Nm	(15)
MODE SENSE	Nm	(1A)
PREVENT/ALLOW MEDIUM REMOVAL	Nm	(1E)
READ	B	(08)
READ BLOCK LIMITS	E	(05)
READ REVERSE	Nm	(0F)
RECOVER BUFFERED DATA	Nm	(14)
RELEASE UNIT	Nm	(17)
RESERVE UNIT	Nm	(16)
REWIND	B	(01)
SPACE	Nm	(11)
TRACK SELECT	Nm	(0B)
VERIFY	Nm	(13)
WRITE	B	(0A)
WRITE FILE MARKS	B	(10)

B = mandatory in Basic Level

E = mandatory in Extended Level

Nm = not mandatory in either level

6.4.1 ERASE Command (19)

Specific command for sequential access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	1	0	0	1
2nd		LUN			0	0	0	0	LONG-Bit
3rd		0	0	0	0	0	0	0	0
4th		0	0	0	0	0	0	0	0
5th		0	0	0	0	0	0	0	0
6th		Private Use		0	0	0	0	Flag	Link

The ERASE command requests the Logical Unit to erase all or part of the medium, starting from the current position. The medium shall be erased or a pattern shall be written on the medium so that it appears as gap to the Target.

When bit 1 of the 2nd byte, the Long Bit, is set to ZERO only a part of the medium shall be erased. The distance to be erased may be defined by the sequential access device.

When the Long Bit is set to ONE, all the medium shall be completely erased from the current position on.

The Target may reject ERASE commands with the Long Bit set to ONE if the medium is not positioned at the beginning of medium.

6.4.2 LOAD/UNLOAD Command (1B)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	0	1	1
2nd	LUN			0	0	0	0	I-Bit
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	RT-Bit	LOAD-Bit
6th	Private Use		0	0	0	0	Flag	Link

The LOAD/UNLOAD command requests the Logical Unit to position the medium for operation or removal.

When bit 1 of the 2nd byte, the Immediate Bit (I-Bit), is set to ONE the COMPLETION status may be returned as soon as the execution of the command is initiated.

When the Immediate Bit is set to ZERO the COMPLETION status shall be returned only when the command is completed (tape at the BOT or LP marker).

When bit 1 of the 5th byte, the Load Bit, is set to ZERO the medium on the Logical Unit shall be unloaded and positioned for removal from the sequential access device.

When the Load Bit is set to ONE the medium on the Logical Unit shall be loaded and positioned to the beginning of medium.

When bit 2 of the 5th byte, the Re-tension Bit (RT-Bit) is set to ONE, the medium on the addressed LUN shall be correctly tensioned before loading or unloading is executed.

6.4.3 MODE SELECT Command (15)

Specific command for sequential access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	0	1
2nd		LUN			0	0	0	0	0
3rd		0	0	0	0	0	0	0	0
4th		0	0	0	0	0	0	0	0
5th		Parameter List Length							
6th		Private Use		0	0	0	0	Flag	Link

The MODE SELECT command provides a means for the Initiator to specify to the Target the parameters of the medium, the Logical Unit or the peripheral device.

The Parameter List Length field specifies the total length in bytes of the Parameter List which shall be transferred to the Target, including Private Use parameter bytes, if any.

If the value specified by the Parameter List Length field equals zero, no data shall be transferred. This condition shall not be considered as an error.

The Parameter List comprises a 4-byte Header followed by zero or more 8-byte Block Descriptors, the last Block Descriptor may be followed by parameter bytes for Private Use.

Header

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	0	0	0
2nd		0	0	0	0	0	0	0	0
3rd		0	Buffered Mode			Speed			
4th		Block Descriptor Length							

Block Descriptor

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		Density Code							
2nd		Number of Blocks (MSB)							
3rd		Number of Blocks							
4th		Number of Blocks (LSB)							
5th		0	0	0	0	0	0	0	0
6th		Block Length (MSB)							
7th		Block Length							
8th		Block Length (LSB)							

Private Use Parameter Bytes

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		Parameter Bytes							
n th		Parameter Bytes							

Bits 7, 6, 5 of the 3rd byte of the Header, the Buffered Mode field, specifies:

If set to (0) : The Target shall report GOOD STATUS on WRITE commands when data is successfully written on the medium.

If set to (1) : The Target may report GOOD STATUS when data has been transferred to the Target buffer. One or more blocks may be buffered prior to writing the blocks on the medium.

The other values of this field, i.e. (2) to (F) are reserved for future standardization and shall not be used.

Bits 4 to 1 of the 3rd byte of the Header, the Speed field specifies:

If set to (0) : Default (default speed of peripheral device)

If set to (1) : Low speed (the lowest speed of the peripheral device)

If set to
(2) to (F) : Increasing peripheral device speeds.

The Block Descriptor Length field specifies the length in bytes of all Block Descriptors. If the number specified by the Block Descriptor Length field equals zero, no Block Descriptors are included in the Parameter List. This shall not be considered as an error condition.

The Block Descriptors specify the characteristics of the medium for all or part of a Logical Unit.

The density Indicator shall specify:

If set to (00) : Default density of the peripheral device

If set to (01) : ECMA-62 - Data Interchange on 12,7 mm 9-Track Magnetic Tape - 32 ftpmm, NRZ1, 32 cpmm

If set to (02) : ECMA-62 - Data Interchange on 12,7 mm 9-Track Magnetic Tape, 126 ftpmm, Phase Encoding, 63 cpmm

If set to (03) : ECMA-62 - Data Interchange on 12,7 mm 9-Track Magnetic Tape, 356 ftpmm, NRZ1, 245 cpmm GCR

If set to (04) to (06): Not specified by this Standard

If set to (07) : ECMA-79 - Data Interchange on 6,30 mm Magnetic Tape Cartridge using MFM Recording at 252 ftpmm

If set to (08) : Not specified by this Standard

If set to (09) : Draft ECMA Standard: 12,7 mm Wide Magnetic Tape Cartridge using IFM Recording on 18 Tracks at 1944 ftpmm, GCR

If set to (0A) : Not specified by this Standard

If set to (0B) : ECMA-46 - Data Interchange on 6,30 mm Magnetic Tape Cartridge (63 bpmm, Phase Encoded)

If set to (0C) to (0D): Not specified by this Standard

If set to (0E) : ECMA-98 - Data Interchange on 6,30 mm Magnetic Tape Cartridge using NRZ1 Recording at 394 ftpmm - Streaming Mode

The values (0F) to (7F) are reserved for future standardization and shall not be used.

The values (80) to (FF) are left for Private Use.

The Number of Blocks field specifies the number of logical blocks on the medium that are to be formatted as specified by the Density Indicator field. If the number specified by the Number of Blocks field is equal to zero all remaining blocks of the Logical Unit shall be formatted.

The Block Length field specifies the block length of the blocks to be formatted. If the value specified by this field equals zero, this indicates that the length is variable.

6.4.4 MODE SENSE Command (1A)

Specific command for sequential access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	0	1	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Allocation Length							
6th	Private Use		0	0	0	0	Flag	Link

The MODE SENSE command provides a means for a Target to report the parameters of its medium, Logical Unit or peripheral device. It is a complementary command to the MODE SELECT command.

The Allocation Length field specifies the total number of bytes that the Initiator has allocated for the returned Mode Sense Data.

If the value specified by the Allocation Length field is equal to zero, no Mode Sense Data shall be transferred. This shall not be considered as an error condition.

The Target shall terminate the command when the number of bytes specified by the Allocation Length field has been transferred or when all available Mode Sense Data has been transferred to the Initiator.

The Parameter List specifying the Mode Sense Data comprises a 4-byte Header, followed by zero or more 8-byte Block Descriptors, the last Block Descriptor may be followed by parameter bytes for Private Use.

Header

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	Sense Data Length							
2nd	Medium Type							
3rd	WP-Bit	Buffered Mode			Speed			
4th	Block Descriptor Length							

The Parameter List specifying the MODE SENSE DATA comprises a 4-byte Header, followed by zero or more 8-byte Block Descriptors, the last Block Descriptor may be followed by parameter bytes for Private Use.

Block Descriptor

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		Density Indicator							
2nd		Number of Blocks (MSB)							
3rd		Number of Blocks							
4th		Number of Blocks (LSB)							
5th		0	0	0	0	0	0	0	0
6th		Block Length (MSB)							
7th		Block Length							
8th		Block Length (LSB)							

Private Use Parameter Bytes

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		Parameter Bytes							
n th		Parameter Bytes							

The 2nd byte, the Medium Type field, specifies:

If set to (00): Default, i.e. only one type of medium is supported.

The values (01) to (7F) are reserved for future standardization and shall not be used. The values (80) to (FF) are left for Private Use.

Bit 1 of the 3rd byte, the Write-Protect Bit (WP-Bit), when set to ZERO, indicates that a write operation on the medium can be performed.

When set to ONE, this Write-Protect Bit indicates that no write operation can be performed.

Bits 7, 6, 5 of the 3rd byte of the Header, the Buffered Mode field, specifies:

If set to (0) : The Target shall report GOOD STATUS on WRITE commands when data is successfully written on the medium.

If set to (1) : The Target may report GOOD STATUS when data has been transferred to the Target buffer. One or more blocks may be buffered prior to writing the blocks on the medium.

The other values of this field, i.e. (2) to (F) are reserved for future standardization and shall not be used.

Bits 4 to 1 of the 3rd byte of the Header, the Speed field specifies:

If set to (0) : Default (default speed of peripheral device)

If set to (1) : Low speed (the lowest speed of the peripheral device)

If set to (2) to (F) : Increasing peripheral device speeds.

The Block Descriptor Length field specifies the total length in bytes of all Block Descriptors. If the number specified by the Block Descriptor Length field is equal to zero, no Block Descriptors are included in the Parameter List. This shall not be considered as an error condition.

The Block Descriptors specify the characteristics of the medium for all or part of a Logical Unit.

The density Indicator shall specify:

If set to (00) : Default density of the peripheral device

If set to (01) : ECMA-62 - Data Interchange on 12,7 mm 9-Track Magnetic Tape - 32 ftpmm, NRZ1, 32 cpmm

If set to (02) : ECMA-62 - Data Interchange on 12,7 mm 9-Track Magnetic Tape, 126 ftpmm, Phase Encoding, 63 cpmm

If set to (03) : ECMA-62 - Data Interchange on 12,7 mm 9-Track Magnetic Tape, 356 ftpmm, NRZ1, 245 cpmm GCR

If set to (04) to (06): Not specified by this Standard

If set to (07) : ECMA-79 - Data Interchange on 6,30 mm Magnetic Tape Cartridge using MFM Recording at 252 ftpmm

If set to (08) : Not specified by this Standard

If set to (09)	: Draft ECMA Standard: 12,7 mm Wide Magnetic Tape Cartridge using IFM Recording on 18 Tracks at 1944 ftpmm, GCR
If set to (0A)	: Not specified by this Standard
If set to (0B)	: ECMA-46 - Data Interchange on 6,30 mm Magnetic Tape Cartridge (63 bpmm, Phase Encoded)
If set to (0C) to (0D)	: Not specified by this Standard
If set to (0E)	: ECMA-98 - Data Interchange on 6,30 mm Magnetic Tape Cartridge using NRZ1 Recording at 394 ftpmm - Streaming Mode

The values (0F) to (7F) are reserved for future standardization and shall not be used.

The values (80) to (FF) are left for Private Use.

The Number of Blocks field specifies the number of logical blocks on the medium that are to be formatted as specified by the Density Indicator field. If the number specified by the Number of Blocks field equals zero all remaining blocks of the Logical Unit shall be formatted.

The Block Length field specifies the block length of the blocks to be formatted. If the value specified by this field equals zero, this indicates that the length is variable.

6.4.5 PREVENT/ALLOW MEDIUM REMOVAL Command (1E)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	1	1	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	Prev-Bit
6th	Private Use		0	0	0	0	Flag	Link

The PREVENT/ALLOW MEDIUM REMOVAL command requests the Target to allow or to prevent the removal of the medium on the Logical Unit.

When bit 1 of the 5th byte, the Prevent Bit (Prev-Bit), is set to ONE, removal of the medium is prevented.

This condition shall remain in effect until a PREVENT/ALLOW MEDIUM REMOVAL command with the Prevent Bit set to ZERO or a RESET BUS DEVICE message is received or a Hard Reset condition is detected.

6.4.6 READ Command (08)

Specific command for sequential access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	1	0	0	0
2nd		LUN			0	0	0	0	Fixed-Bit
3rd		Transfer Length (MSB)							
4th		Transfer Length							
5th		Transfer Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The READ command requests the Target to transfer one or more blocks to the Initiator beginning with the next block on the Logical Unit.

When bit 1 of the 2nd byte, the Fixed Bit, is set to ZERO, a single block shall be transferred, whereby the value specified by the Transfer Length field indicates the maximum number of bytes the Initiator has allocated for the returned data.

If the block length is different from the value specified by the Transfer Length field, the CHECK CONDITION status shall be sent to the Initiator and the ILI-Bit (Illegal Length Indicator) and the Validity Bit shall be set to ONE in the Extended Sense Bytes. The Information Bytes in the Extended Sense Bytes field shall be set to the difference between the requested value of the Transfer Length field and the actual block length. Targets which do not support negative values of this difference shall set the ILI-Bit to ONE and the Information Bytes to ZERO when the actual block length is larger than the value specified by the Transfer Length field. The medium shall be positioned after the block (end of medium side).

When the Fixed Bit is set to ONE the value of the Transfer Length field specifies the number of blocks transferred to the Initiator. This is valid only if the Logical Unit is currently operating in the fixed-length mode.

The Logical Unit is in fixed-length mode when either of the following conditions are true:

- The Logical Unit reports the same value for the minimum block length and the maximum block length in response to the READ BLOCK LIMITS command. In this case the current block length is the value returned.

- The Logical Unit has been instructed to use fixed-length blocks with the MODE SELECT command. In this case the current block length is the block length defined in the MODE SENSE command.

If the Logical Unit is not in fixed-length mode, then it is in variable-length mode and the Target shall reject a READ command with the Fixed Bit set to ONE by returning CHECK CONDITION status and by setting the Sense Key to ILLEGAL REQUEST.

A successful READ command with the Fixed Bit set to ONE shall transfer the current block length multiplied by the value specified by the Transfer Length field to the Initiator. Upon termination of the READ command the medium shall be positioned after the last block transferred (end of medium side).

If the Fixed Bit is set to ONE and if a block is read the length of which is larger or smaller than the current block length, CHECK CONDITION status shall be returned to the Initiator. The ILI-Bit and the Validity-Bit in the Extended Sense Bytes shall be set to ONE. The Information Bytes field shall be set to the difference between the requested value of the Transfer Length field and the actual number of blocks read (not including the incorrect length block). Upon termination of the READ command the medium shall be positioned after the incorrect length block (end of medium side).

If the Logical Unit reads a File Mark during a READ command it shall send a CHECK CONDITION status to the Initiator and set the FM Bit to ONE in the Extended Sense Bytes. Upon termination, the medium shall be positioned after the File Mark (end of medium side).

If the Fixed Bit is set to ONE in the READ command the Target shall set the Validity Bit to ONE and the Information Bytes field shall be set to the difference between the requested value of the Transfer Length field and the actual number of blocks (not including the File Mark).

If a Logical Unit detects the physical end of medium during a READ command the Target shall return a CHECK CONDITION status to the Initiator and shall set the EOM Bit to ONE in the Extended Sense Bytes. The Sense Key shall be set to MEDIUM ERROR. If the Fixed Bit is set to ONE in the READ command the Target shall set the Validity Bit to ONE and the Information Bytes field to the difference between the requested value specified by the Transfer Length field and the actual number of blocks successfully read. The medium position following this condition is not defined.

When the value specified by the Transfer Length field equals zero, no data shall be transferred and the current position on the Logical Unit shall not be changed. This condition shall not be considered as an error.

6.4.7 READ BLOCK LIMITS Command (05)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	1	0	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

This command requests the block length limits for the Logical Unit to be returned. The following read block Limits Data shall be sent during the DATA IN phase of the command.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	Maximum Block Length (MSB)							
3rd	Maximum Block Length							
4th	Maximum Block Length (LSB)							
5th	Minimum Block Length (MSB)							
6th	Minimum Block Length (LSB)							

If the value specified by the Maximum Block Length field equals that specified by the Minimum Block Length field, fixed-length blocks are specified.

If the value specified by the Maximum Block Length field and that specified by the Minimum Block Length field are different, variable-length blocks are specified.

The upper limit will not be specified if the value specified by the Maximum Block Length field equals zero and that specified by the Minimum Block Length field does not equal zero.

6.4.8 READ REVERSE Command (0F)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	1	1	1
2nd	LUN			0	0	0	0	Fixed-Bit
3rd	Transfer Length (MSB)							
4th	Transfer Length							
5th	Transfer Length (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The READ REVERSE command is equivalent to the READ command except that the medium motion is in the reverse direction. Thus, the blocks and bytes within the blocks are transferred in the reverse order and upon termination, the medium is positioned before the last block (beginning of medium side).

This command shall terminate with CHECK CONDITION status and the EOM Bit set to ONE in the Extended Sense Bytes if the Beginning of Medium marker (BOT) or the Load Point marker (LP) is encountered.

File Mark handling is the same as in the READ command except that the medium position upon command termination shall be before the File Mark (beginning of medium side).

If the value specified by the Transfer Length field equals zero no data shall be transferred and the current position on the Logical Unit shall not be changed. This condition shall not be considered as an error.

6.4.9 RECOVER BUFFERED DATA Command (14)

Specific command for sequential access devices.

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	0	0
2nd		LUN			0	0	0	0	Fixed-Bit
3rd		Transfer Length (MSB)							
4th		Transfer Length							
5th		Transfer Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The RECOVER BUFFERED DATA command is used to read data that has been transferred to a Target buffer but has not been written on the medium. It is normally used to recover from error or exception conditions that make it impossible to write the buffered data on the medium.

The command is similar to the READ command except that the data is transferred from the Target buffer instead of the medium. The order in which blocks are transferred is the same as if they would have been transferred to the medium.

One or more RECOVER BUFFERED DATA commands may be used to read the unwritten buffered data. If an attempt is made to recover more logical blocks of data than are contained in the Target buffer the command shall be terminated with a CHECK CONDITION status. The EOM Bit shall be set to ONE in the Extended Sense Bytes. If the Fixed Bit is set to ONE, the Validity Bit shall be set to ONE in the Extended Sense Bytes and the value of the Information Bytes field shall be set to the difference between the requested value of the Transfer Length field and the actual number of blocks transferred.

The value specified by the Transfer Length field indicates the number of contiguous blocks of data to be transferred. A value of the Transfer Length field equal to zero indicates that no data shall be transferred. This condition shall not be considered as an error.

6.4.10 RELEASE Command (17)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	1	1	1
2nd	L U N			3rdP-Bit	3rd Party Device ID			0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

The RELEASE command shall release the Logical Unit if it is currently reserved to the requesting Initiator.

It is not an error to attempt to release a Logical Unit which is not currently reserved to the requesting Initiator and it shall not be released.

If bit 5 of the 2nd byte, the Third Party Bit (3rdP-Bit) is set to ONE, the Logical Unit shall be released only if the reservation was made by the same Initiator and with the same Bus Device specified in the Third Party ID field.

If the Third Party Bit is set to ONE and the Third Party Reservation is not implemented in the Target, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

6.4.11 RESERVE UNIT Command (16)

Specific command for sequential access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	1	1	0
2nd	L U N			3rdP-Bit	3rd Party Device ID			0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

The RESERVE UNIT command shall reserve the specified Logical Unit for the exclusive use by the requesting Initiator.

The reservation shall remain in effect until a RELEASE UNIT command is received from the same Initiator, or a RESET BUS DEVICE message is received from any Initiator or a Hard Reset condition is detected. The occurrence of the RESET BUS DEVICE message or the Hard Reset condition is indicated by a Sense Key set to UNIT ATTENTION on the next command following the condition. It is not an error to issue this command to a Logical Unit which is currently reserved to the requesting Initiator.

If a command is received from any other Initiator while the Logical Unit is reserved to an Initiator, the Target shall either queue the command and disconnect until the Logical Unit is released, or terminate the command with RESERVATION CONFLICT status.

If bit 5 of the 2nd byte, the Third Party Bit (3rdP-Bit) is set to ONE, the Logical Unit shall be reserved for the Bus Device specified in the Third Party Device ID field. The reservation shall be in effect until it is released by the same Initiator or by the RESET BUS DEVICE message or a Hard Reset condition. The Target shall ignore any attempt to release the reservation from any other Initiator.

If the Third Party Bit is set to ONE and the Third Party Reservation is not implemented in the Target, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

6.4.12 REWIND Command (01)

Specific command for sequential access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	1
2nd	LUN			0	0	0	0	I-Bit
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

The REWIND command requests the Target to rewind the medium of the Logical Unit to the Beginning of Medium marker or to the Load Point marker.

If bit 1 of the 2nd byte, the Immediate Bit (I-Bit) is set ot ONE the COMPLETION status may be returned as soon as the execution of the command is initiated.

If the Immediate Bit is set to ZERO the COMPLETION status shall be returned when the command is completed (tape at Beginning of Medium marker or at Load Point marker).

6.4.13 SPACE Command (11)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	0	0	1
2nd	LUN			0	0	0	Search	
3rd	Count (MSB)							
4th	Count							
5th	Count (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The SPACE command provides a number of positioning functions which are determined by the Search field and the Count field.

Some Bus Devices may support only a subset of the functions of this command and such devices shall return CHECK CONDITION status and set the Sense Key to ILLEGAL REQUEST in response to functions not supported.

Bits 1 and 2 of the 2nd byte, the Search field, specify:

Search Blocks : 00
Search File Marks : 01
Search Sequential File Marks : 10
Search Physical end of data : 11

When spacing over blocks or File Marks, the Count field specifies the number of blocks or File Marks to be spaced over. A positive value N in the Count field shall cause a forward movement of the medium over N blocks or File Marks ending on the end-of-medium side of the N-th block or File Mark.

If the value specified by the Count field equals zero, no medium movement shall occur. A negative value of N (two's complement notation) in the Count field shall cause a reverse movement of the medium over N blocks or File Marks ending on the beginning-of-medium side of the N-th block or File Mark.

If a File Mark is encountered while spacing over blocks, the medium movement shall be stopped. If the movement was in the forward direction the medium shall be positioned on the end-of-medium side of the File Mark and if the movement was in the reverse direction the medium shall be positioned on the beginning-of-medium side of the File Mark.

CHECK CONDITION status shall be sent to the Initiator and the File Mark Bit and Validity Bit in the Extended Sense Bytes shall be set to ONE. The Information Bytes field shall be set to the difference between the requested value in the Count field and the actual number of blocks spaced over (not including the File Mark).

If the physical end of the medium is encountered while spacing over blocks or File Marks in the forward direction, the Target shall return a CHECK CONDITION status to the Initiator and shall set the EOM Bit to ONE in the Extended Sense Bytes. The Sense Key shall be set to MEDIUM ERROR. The Validity Bit shall be set to ONE and the information bytes shall be set to the difference (residue) between the requested value in the Count field and the actual number of blocks or File Marks spaced over.

If the Beginning-of-Medium marker (BOT) or Load Point (LP) marker is encountered while spacing over blocks or File Marks in the reverse direction, the Target shall return a CHECK CONDITION status to the Initiator and shall set the EOM Bit to ONE in the Extended Sense Bytes. The Sense Key shall be set to NO SENSE. The Validity Bit shall be set to ONE and the Information Bytes field shall be set to the difference between the requested value in the Count field and the actual number of blocks or File Marks spaced over.

When the Search field is set to 10, i.e. for searching sequential File Marks the value in the Count field shall be interpreted as follows.

A positive value N shall cause a forward movement of the medium to the first occurrence of N or more consecutive File Marks stopping on the end-of-medium side of the N-th File Mark.

If the value specified by the Count field equals zero, no movement shall occur.

A negative value of N (two's complement notation) shall cause a reverse movement of the medium to the first occurrence of N or more consecutive File Marks stopping on the beginning-of-medium side of the N-th File Mark.

When spacing to the physical end of data, the Count field is ignored. The medium shall be moved in the forward direction until the Logical Unit encounters physical end of data as defined for the actual sequential access device. For some sequential access devices the physical end of data may be defined as an erased area on the medium but other definitions are not precluded. Targets which implement this function shall leave the medium positioned such that a subsequent WRITE command would append data to the last recorded information on the medium.

6.4.14 TRACK SELECT Command (0B)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	0	1	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Track Number							
6th	Private Use		0	0	0	0	Flag	Link

The TRACK SELECT command requests that the track specified by the Track Number field be selected.

6.4.15 VERIFY Command (13)

Specific command for sequential access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	0	1	1
2nd		LUN			0	0	0	BCom-Bit	Fixed-Bit
3rd		Verification Length (MSB)							
4th		Verification Length							
5th		Verification Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The VERIFY command verifies one or more blocks, beginning with the next block on the Logical Unit.

If bit 2 of the 2nd byte, the Byte Compare Bit (BCom-Bit) is set to ZERO, the medium is checked for read errors (CRC, ECC, etc.) and no data is transferred between the Initiator and Target.

If the Byte Compare Bit is set to ONE, the data on the medium and the data transferred from the Initiator are compared byte by byte. Data shall be transferred from the Initiator to the Target as in the WRITE command.

If bit 1 of the 2nd byte, the Fixed Bit, is set to ZERO, the next block from the Logical Unit shall be verified. The value specified by the Verification Length field indicates the number of bytes to verify.

If the Fixed Bit is set to ONE, a number of blocks beginning with the next block from the Logical Unit shall be verified. The value specified by the Verification Length field indicates the number of blocks to verify. This form of the command is only valid if the Logical Unit is currently working in fixed-block mode as defined in the READ command.

If a data miscomparison occurs the command shall terminate with a CHECK CONDITION status and the Sense Key set to MISCOMPARE. If the Fixed Bit is set to ONE the Validity Bit shall be set to ONE in the Extended Sense Bytes and the Information Bytes field to the difference between the value specified by the Verification Length field and the actual number of blocks successfully verified. The medium shall be positioned after the block containing the miscomparison (end of medium side).

The VERIFY command terminates when the number of blocks verified equals the value specified by the Verification Length field, or when a File Mark is encountered or when the physical end of medium is encountered. The status and Sense Data for each of these conditions shall be handled as in the READ command. Upon completion of the VERIFY command the medium shall be positioned after the last block from which data was verified or after a File Mark, if encountered.

When the value specified by the Verification Length field equals zero, no data shall be verified and the current position on the Logical Unit shall not be changed. This condition shall not be considered as an error.

6.4.16 WRITE Command (0A)

Specific command for sequential access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	1	0	1	0
2nd		LUN			0	0	0	0	Fixed-Bit
3rd		Transfer Length (MSB)							
4th		Transfer Length							
5th		Transfer Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The WRITE command transfers one or more blocks from the Initiator to the current position of the medium of the Logical Unit. Bit 1 of the 2nd byte, the Fixed Bit, specifies both the meaning of the value specified by the Transfer Length field and whether fixed-length or variable-length blocks shall be transferred.

If the Fixed Bit is set to ZERO, a single block shall be transferred from the Initiator and shall be written on the medium of the Logical Unit beginning at its current position. The value specified by the Transfer Length field indicates the length (in bytes) of the block to be written. The requested block length must be within the minimum and maximum length range (as returned by the READ BLOCK LIMITS command). If this condition is not met, a CHECK CONDITION status shall be returned, the Sense Key shall be set to ILLEGAL REQUEST and no data shall be written. Upon successful termination, the medium shall be positioned after the block written by this command (end of medium side).

If the Fixed Bit is set to ONE, the value specified by the Transfer Length field indicates the number of blocks to be transferred to the Logical Unit beginning at the current medium position. This WRITE command is valid only if the Logical Unit is currently operating in fixed-length mode (see READ command). Upon termination, the medium shall be positioned after the block written by this command (end of medium side).

If the end-of-medium marker is encountered while writing, an attempt to finish writing any buffered data may be made. The WRITE command shall terminate with CHECK CONDITION status and the EOM Bit set to ONE in the Extended Sense Bytes. If any data remains in the Target's buffer, the Sense Key shall be set to VOLUME OVERFLOW.

If the Fixed Bit is set to ONE and if the Logical Unit is not buffered (BUFFERED MODE in the MODE SELECT command is ZERO), then the Validity Bit in the Extended Sense Bytes shall be set to ONE and the Information Bytes field shall specify the difference (residue) between the requested transfer length and the actual number of blocks written to the medium.

If the Fixed Bit is set to ONE and if the Logical Unit is buffered (BUFFERED MODE in the MODE SELECT command is ONE), the Validity Bit shall be set to ONE in the Extended Sense Bytes and the Information Bytes field shall specify the total number of blocks not written (the number of blocks not transferred from the Initiator plus the number of blocks remaining in the buffer of the Target). The value specified by the Information Bytes field may in this case exceed that specified by the Transfer Length field.

When the value specified by the Transfer Length field equals zero, no data shall be transferred and the current position on the Logical Unit shall not be changed. This condition shall not be considered as an error.

6.4.17 WRITE FILE MARK Command (10)

Specific command for sequential access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	0	0	0
2nd	LUN			0	0	0	0	0
3rd	Number of File Marks (MSB)							
4th	Number of File Marks							
5th	Number of File Marks (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The WRITE FILE MARK command causes the number of File Marks specified by the Number of File Marks field to be written beginning at the current medium position on the Logical Unit.

This command is also used to force any buffered data (see Buffered Mode field in the MODE SELECT command) to be written. This command shall not return GOOD status unless all buffered data blocks and the File Marks are correctly written on the medium.

If the end-of-medium marker is encountered while writing, an attempt to finish writing any buffered data may be made. The command shall terminate with CHECK CONDITION status and the EOM Bit set to ONE in the Extended Sense Bytes. If any File Marks remain to be written then the Sense Key shall be set to VOLUME OVERFLOW.

When the Logical Unit is not in buffered mode (Buffered Mode field in the MODE SELECT command set to ZERO) the Validity Bit in the Extended Sense bytes shall be set to ONE and the value specified by the Information Bytes field shall indicate the number of unwritten File Marks.

When the Logical Unit is in buffered mode (Buffered Mode field in the MODE SELECT command set to ONE), the Validity Bit shall be set to ONE and the value specified by the Information Bytes field shall indicate the total number of blocks not written (number of blocks remaining in the Target's buffer and the number of unwritten File Marks). In this last case the value specified by the Information Bytes field may exceed that recorded in the Transfer Length field.

If the value specified by the Number of File Marks field equals zero, no File Marks shall be written and the current position of the medium shall not be changed. This condition shall not be considered as an error.

6.5 Command Description for Printers

Device-Dependent Commands

Name		Identifier
FLUSH BUFFER	Nm	(10)
FORMAT	Nm	(04)
MODE SELECT	Nm	(15)
MODE SENSE	Nm	(1A)
PRINT	B	(0A)
RECOVER BUFFERED DATA	Nm	(14)
RELEASE	Nm	(17)
RESERVED UNIT	Nm	(16)
SLEW AND PRINT	Nm	(0B)
STOP PRINT	Nm	(1B)

B = mandatory in Basic Level

Nm = not mandatory in either level

6.5.1 FLUSH BUFFER Command (10)

Specific command for printer devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	0	0	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	0	0	0	0	0	0	0	0
6th	Private Use		0	0	0	0	Flag	Link

The FLUSH BUFFER command provides a means for an Initiator to ensure that the data has been successfully printed prior to releasing the printer.

When all buffered data is actually printed the command shall be terminated with a GOOD status. If it is not possible to finish printing all of the buffered data (due to printer error), then the command shall be terminated with a CHECK CONDITION status and the appropriate Sense Key.

6.5.2 FORMAT Command (04)

Specific command for printer devices.

BYTE ↓	BIT →	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	1	0	0
2nd		LUN			0	0	0	Format Type	
3rd		Transfer Length (MSB)							
4th		Transfer Length							
5th		Transfer Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The FORMAT command provides a means to specify forms or fonts to printers that support programmable forms or fonts. The format information sent is not specified by this Standard as it is dependent on the printer used.

The Format Type field specifies the type of format information to be transferred from the Initiator to the Target. This field is defined as follows:

Bit 2	Bit 1	Format Type
0	0	Set Form
0	1	Set Font
1	0	Private Use
1	1	Reserved for future standardization

The Transfer Length field specifies the length in bytes of format information that is to be sent by the Initiator during the DATA OUT phase. If the value specified by the Transfer Length field equals zero, no format information shall be sent. This shall not be considered as an error condition.

6.5.3 MODE SELECT Command (15)

Specific command for printer devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	1	0	1
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Length of Parameter List							
6th	Private Use		0	0	0	0	Flag	Link

The MODE SELECT command provides a means to specify parameters of the medium, the Logical Unit or the printer.

The Length of Parameter List field specifies the length in bytes of an associated Parameter List which is sent as data during the DATA OUT phase. If the value specified by the Length of Parameter List field equals zero, no data shall be sent. This shall not be considered as an error condition.

The Parameter List contains a 4-byte Header, followed by the parameters, if any.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	0	0	0	0
2nd	0	0	0	0	0	0	0	0
3rd	0	Buffered Mode			0	0	0	0
4th	0	0	0	0	0	0	0	0

Parameters

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Parameter							
nth	Parameter							

The Buffered Mode field specifies:

(0) The Logical Unit shall not report GOOD status on a PRINT command or a SLEW AND PRINT command until the data is actually printed on the medium.

(1) The Logical Unit may report GOOD status on a PRINT command or a SLEW AND PRINT command as soon as the data is transferred to the Target's data buffer. One or more commands may be buffered prior to actual printing.

(2) - (7) Reserved for future standardization.

6.5.4 MODE SENSE Command (1A)

Specific command for printer devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	0	1	0
2nd	LUN			0	0	0	0	0
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Allocation Length							
6th	Private Use		0	0	0	0	Flag	Link

The MODE SENSE command provides a means for a Target to report the parameters of its medium, Logical Unit, or peripheral device. It is a complementary command to the MODE SELECT command.

The Allocation Length field specifies the total number of bytes that the Initiator has allocated for returned Sense Data. If the value specified by the Allocation Length field equals zero, no Sense Data shall be transferred. This shall not be considered an error condition. Any other value shall indicate the maximum number of bytes to be transferred. The Target shall terminate the DATA IN phase when this number of bytes has been transferred or when all available Mode Sense Data has been transferred to the Initiator.

The Sense Data comprise a 4-byte Header, followed by the Private Use parameters, if any.

Header

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	Sense Data Length							
2nd	0	0	0	0	0	0	0	0
3rd	0	Buffered Mode			0	0	0	0
4th	0	0	0	0	0	0	0	0

Parameters

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		Parameter							
n th		Parameter							

The Sense Data Length field specifies the length in bytes of the data available to be transferred during the DATA IN phase.

The Buffered Mode field specifies:

(0) The Target does not report GOOD status on a PRINT command or a SLEW AND PRINT command until the data is actually printed on the medium.

(1) The Target may report GOOD status on PRINT command or SLEW AND PRINT command as soon as the data is transferred to the Target data buffer. One or more commands may be buffered prior to actual printing.

(2) - (7) Reserved for future standardization.

6.5.5 PRINT Command (0A)

Specific command for printer devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	1	0	1	0
2nd		LUN			0	0	0	0	0
3rd		Transfer Length (MSB)							
4th		Transfer Length							
5th		Transfer Length (LSB)							
6th		Private Use		0	0	0	0	Flag	Link

The PRINT command requests the Target to print the specified number of data bytes.

The Transfer Length field specifies the length in bytes of data that is to be sent during the DATA OUT phase. If the value specified by the Transfer Length field equals zero, no data shall be sent. This shall not be considered as an error condition.

6.5.6 RECOVER BUFFERED DATA Command (14)

Specific command for printer devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	0	1	0	0
2nd	LUN			0	0	0	0	0
3rd	Transfer Length (MSB)							
4th	Transfer Length							
5th	Transfer Length (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The RECOVER BUFFERED DATA command requests the Target to return to the Initiator data that has not been printed.

This command is normally used only to recover from an error that makes it impossible to print the buffered data. The order in which the data is transferred from the Target to the Initiator is the same as it was when it was previously transferred using the PRINT command or SLEW AND PRINT command. Data that is transferred by this command is deleted from the Target data buffer.

One or more RECOVER BUFFERED DATA commands may be used to return the unprinted buffered data. If an attempt is made to recover more data than is contained in the buffer, the command shall be terminated with a CHECK CONDITION status and the Sense Key set to NO SENSE. In addition, the EOM Bit, the Validity Bit and the ILI Bit in the Extended Sense Bytes shall be set to ONE. The Information Bytes field shall be set to the difference between the value specified by the Transfer Length field and the actual number of bytes returned.

The Transfer Length field specifies the length in bytes of data that is to be transferred during the DATA IN phase. If the value specified by the Transfer Length field equals zero, no data shall be sent. This shall not be considered as an error condition.

6.5.7 RELEASE Command (17)

Specific command for printer devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	1	1
2nd		L U N			3rdP-Bit	3rd Party Device ID			0
3rd		0	0	0	0	0	0	0	0
4th		0	0	0	0	0	0	0	0
5th		0	0	0	0	0	0	0	0
6th		Private Use		0	0	0	0	Flag	Link

The RELEASE command shall release the Logical Unit if it is currently reserved by the requesting Initiator.

It is not an error to attempt to release a Logical Unit which is not currently reserved by the requesting Initiator; however, it shall not be released if reserved by another Initiator.

The Third Party Release option allows an Initiator to release a Logical Unit that was previously reserved using Third Party Reservation (see the RESERVE command). This option is intended for use in multiple-Initiator systems which use the COPY command.

If bit 5 of the 2nd byte, the Third Party Bit (3rdP-Bit) is set to ZERO, the Third Party Release is not requested. If the Third Party Bit is set to ONE, the specified Logical Unit is released, but only if the reservation was made using Third Party Reservation by the same Initiator for the same Bus Device as specified by the Third Party Device ID field.

If the Third Party Bit is set to ONE and if the Third Party Reservation is not implemented, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

6.5.8 RESERVE Command (16)

Specific command for printer devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	1	0
2nd		LUN			3rdP-Bit	3rd Party Device ID			0
3rd		0	0	0	0	0	0	0	0
4th		0	0	0	0	0	0	0	0
5th		0	0	0	0	0	0	0	0
6th		Private Use		0	0	0	0	Flag	Link

The RESERVE command shall reserve the specified Logical Unit for the exclusive use by the requesting Initiator.

The reservation shall be in effect until a RELEASE command is received from the same Initiator, or a BUS DEVICE RESET message is received from any Initiator, or a Hard Reset condition occurs. The occurrence of the last two conditions is indicated by a Sense Key set to UNIT ATTENTION on the next command following the condition. It is not an error to issue this command to a Logical Unit which is currently reserved to the requesting Initiator.

If, while a Logical Unit is reserved by an Initiator, a command is received from any other Initiator, a Target shall, at its option, either:

- terminate the command with RESERVATION CONFLICT status, or
- queue the command and disconnect until the Logical Unit is released.

Third Party Reservation option allows an Initiator to reserve a Logical Unit for another Bus Device. This option is intended for use in multiple-Initiator systems which implement the COPY command. Any Target which implements Third Party Reservation shall also implement the Third Party Release option (see the RELEASE command).

If bit 5 of the 2nd byte, the Third Party Bit (3rdP-Bit) is set to ZERO, the Third Party Reservation is not requested. If this bit is set to ONE, the specified Logical Unit is reserved for the exclusive use of the Bus Device specified in the Third Party Device ID field. The reservation shall be in effect until it is released by the same Initiator (or by the BUS DEVICE RESET message

or a Hard Reset condition). The Target shall ignore any attempt to release the reservation made by any other Initiator.

If the Third Party Bit is set to ONE, and the Third Party Reservation option is not implemented, then the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

6.5.9 SLEW AND PRINT Command (0B)

Specific command for printer devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	0	1	1
2nd	LUN			0	0	0	0	Ch-Bit
3rd	Slew Value							
4th	Transfer Length (MSB)							
5th	Transfer Length (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

The SLEW AND PRINT command requests the Target to print the specified number of bytes, after having advanced the paper.

This command is provided for printers that do not support form control information imbedded in the data to be printed.

The Transfer Length field specifies the length in bytes of data to be sent during the DATA OUT phase. If the value specified by the Transfer Length field equals zero, no data shall be sent. This shall not be considered as an error condition.

If bit 1 of the 2nd byte, the Channel Bit (Ch-Bit) is set to ZERO, the 3rd byte, the Slew Value field, specifies the number of lines the form shall be advanced before printing. A value of 255 indicates that the form shall be advanced to the first line of the next form before printing.

If the Channel Bit is set to ONE, the Slew Value field specifies the number of the form control channel to which the form shall be advanced before printing the data.

If the Channel Bit is set to ONE, and the Channel option is not implemented, the command shall be terminated with a CHECK CONDITION status and the Sense Key set to ILLEGAL REQUEST.

6.5.10 STOP PRINT Command (1B)

Specific command for printer devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	1	0	1	1
2nd		LUN			0	0	0	0	Retain-Bit
3rd		Private Use							
4th		0	0	0	0	0	0	0	0
5th		0	0	0	0	0	0	0	0
6th		Private Use		0	0	0	0	Flag	Link

The STOP PRINT command is used to halt printing on buffered devices in an orderly fashion.

If bit 1 of the 2nd byte, the Retain Bit, is set to ZERO, it indicates that the Target buffer shall be discarded. If this bit is set to ONE, the unprinted data is retained. The unprinted data may be recovered by use of the RECOVER BUFFERED DATA command, if implemented. A subsequent PRINT command or SLEW AND PRINT command shall cause the remaining unprinted and unrecovered data to be printed followed by the data transferred by the subsequent command. The point at which printing is suspended by this command is depending on the specific printer in use and is not defined by this Standard.

6.6 Command Description for Processor Devices

Name		Identifier
RECEIVE	Nm	(08)
SEND	B	(0A)

B = mandatory in Basic Level

Nm = not mandatory in either level

The LUN has no specific meaning when applied to processor devices. Its interpretation is left open to the system software.

6.6.1 RECEIVE Command (08)

Specific command for processor devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	0	0	0
2nd	LUN			0	0	0	0	0
3rd	Allocation Length (MSB)							
4th	Allocation Length							
5th	Allocation Length (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

This command requests the Target to transfer data to the Initiator.

The Allocation Length field specifies the number of bytes that the Initiator has allocated for returned data. If the value specified by the Allocation Length field equals zero, no data will be transferred. This shall not be considered as an error condition.

The Target shall terminate the DATA IN phase when the number of bytes specified by the Allocation Length field has been transferred or when all available data has been transferred to the Initiator.

6.6.2 SEND Command (0A)

Specific command for processor devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	0	1	0
2nd	LUN			0	0	0	0	0
3rd	Transfer Length (MSB)							
4th	Transfer Length							
5th	Transfer Length (LSB)							
6th	Private Use		0	0	0	0	Flag	Link

This command requests the Target to transfer data from the Initiator. The Transfer Length field specifies the number of data bytes. If the value specified by the Transfer Length field equals zero, no data will be sent. This shall not be considered as an error condition.

6.7 Command Description for Optical Devices

This clause specifies the commands used for optical Read Only direct access devices and for optical Write-Once-Read-Multiple devices.

6.7.1 Read-Only optical direct-access devices

Name		Identifier
READ	Nm	(08)
READ EXTENDED	B	(28)
VERIFY	Nm	(2F)
MODE SELECT	Nm	(15)
MODE SENSE	Nm	(1A)

B = mandatory in Basic Level

Nm = not mandatory in either level

6.7.1.1 READ Command (08)

Specific command for optical direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	1	0	0	0
2nd		L U N			Logical Block Address (MSB)				
3rd		Logical Block Address							
4th		Logical Block Address (LSB)							
5th		Transfer Length							
6th		Private Use		0	0	0	0	Flag	Link

The READ command requests that the Target transfer data to the Initiator.

The Logical Block Address field specifies the address of the logical block where the read operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value specified by the Transfer Length field equals zero, this means that 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks to be transferred.

The most recent data value written shall be returned.

If an attempt is made to read a blank or previously written block, the command shall be terminated with a CHECK CONDITION status and, if the EXTENDED SENSE command is implemented, the Sense Key shall be set to BLANK CHECK.

6.7.1.2 READ EXTENDED Command (28)

Specific command for optical direct access devices.

BYTE ↓ BIT →	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	0	0	0
2nd	LUN			0	0	0	0	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The READ EXTENDED command requests that the Target transfer data to the Initiator.

Bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit), is set to ONE to indicate that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

The Logical Block Address field specifies the address of the logical block where the read operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value specified by the Transfer Length field equals zero, no logical blocks shall be transferred. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be transferred.

The most recent data value written shall be returned.

If an attempt is made to read a blank or previous written block, the command shall be terminated with a CHECK CONDITION status and, if the EXTENDED SENSE command is implemented, the Sense Key shall be set to BLANK CHECK.

6.7.1.3 VERIFY Command (2F)

Specific command for optical direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	1	0	1	1	1	1
2nd		LUN			0	0	0	BC-Bit	RA-Bit
3rd		Logical Block Address (MSB)							
4th		Logical Block Address							
5th		Logical Block Address							
6th		Logical Block Address (LSB)							
7th		0	0	0	0	0	0	0	0
8th		Verification Length (MSB)							
9th		Verification Length (LSB)							
10th		Private Use		0	0	0	0	Flag	Link

The VERIFY command requests that the Target verify the data written on the medium.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) is set to ONE it indicates that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requies that a previous command in the linked group has accessed a block of data on the Logical Unit.

When bit 2 of the 2nd byte, the Byte Check Bit (BC-Bit), is set to ZERO, the verification is simply a medium verification (CRC, ECC, etc.). When this Byte Check Bit is set to ONE, the verification is a byte-by-byte comparison of data on the medium and the data transferred from the Initiator.

The Logical Block Address field specifies the address of the logical block where the verify operation shall begin.

The Verification Length field specifies the number of contiguous logical blocks of data to be verified. If the value specified by the Transfer Length field equals zero, no logical blocks shall be verified. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be verified.

If an attempt is made to read a blank or previously written block, the command shall be terminated with a CHECK CONDITION status and, if the EXTENDED SENSE command is implemented, the Sense Key shall be set to BLANK CHECK.

6.7.1.4 MODE SELECT Command (15)

Specific command for optical direct-access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	1	0	1	0	1
2nd		LUN			0	0	0	0	0
3rd		0	0	0	0	0	0	0	0
4th		0	0	0	0	0	0	0	0
5th		Parameter List Length							
6th		Private Use		0	0	0	0	Flag	Link

The MODE SELECT command provides a means for the Initiator to specify medium, logical unit, or peripheral device parameters to the Target.

The Parameter List Length field specifies the length in bytes of the MODE SELECT Parameter List that shall be transferred from the Initiator to the Target. If the value specified by the Parameter List Length field equals zero, this indicates that no data shall be transferred. This condition shall not be considered as an error.

The MODE SELECT Parameter List consists of a four-byte Header, followed by zero or more eight-byte Block Descriptors, followed by Private Use parameters, if any.

Header

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	0	0	0
2nd		0	0	0	0	0	0	0	0
3rd		0	0	0	0	0	0	0	EBC-Bit
4th		Block Descriptor Length							

Block Descriptor

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	0	0	0
2nd		Number of Blocks (MSB)							
3rd		Number of Blocks							
4th		Number of Blocks (LBS)							
5th		0	0	0	0	0	0	0	0
6th		Block Length (MSB)							
7th		Block Length							
8th		Block Length (LSB)							

Private Use Parameter(s)

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st									
n th									

If the first bit of the 3rd byte, the Enable Blank Check Bit (EBC) is set to ZERO, blank checking of the medium during write operations is disabled. If the EBC Bit is set to ONE blank checking is enabled. If a non-blank block is found during a write operation, the command shall be terminated with a CHECK CONDITION status and the Sense Key shall be set to BLANK CHECK. For Read-Only direct-access devices, the EBC Bit shall be set to ZERO.

The Block Descriptor Length field specifies the total length in bytes of all the Block Descriptors. It is equal to the number of Block Descriptors times eight. If the value specified by the Block Descriptor Length field equals zero, this indicates that no Block Descriptors shall be included in the Parameter List. This condition shall not be considered as an error.

Each Block Descriptor specifies the medium characteristics for all or part of a Logical Unit. Each Block Descriptor contains a NUMBER OF BLOCKS field and a BLOCK LENGTH field. The NUMBER OF BLOCKS field specifies the number of logical blocks to be formatted with the block length specified in the Block Descriptor. The BLOCK LENGTH field specifies the length in bytes of the logical block to be formatted.

6.7.1.5 MODE SENSE Command (1A)

Specific command for optical direct-access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	1	1	0	1	0
2nd	LUN							
3rd	0	0	0	0	0	0	0	0
4th	0	0	0	0	0	0	0	0
5th	Allocation Length							
6th	Private Use		0	0	0	0	Flag	Link

The MODE SENSE command provides a means for a Target to report its medium, logical unit, or peripheral device parameters to the Initiator. It is a complementary command to the MODE SELECT command for support of a medium that may contain multiple block lengths.

The Allocation Length field specifies the number of bytes that the Initiator has allocated for returned MODE SENSE Data. If the length specified by the Allocation Length field equals zero, no MODE SENSE data shall be transferred. This condition shall not be considered as an error. Any other value indicates the maximum number of bytes that shall be transferred. The Target shall terminate the DATA IN phase when the number of bytes specified by the Allocation Length field has been transferred or when all available MODE SENSE data have been transferred to the Initiator, whichever is less.

The MODE SENSE Data consists of a four-byte Header, followed by zero or more eight-byte Block Descriptors, followed by Private Use parameters, if any.

Header

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		Sense Data Length							
2nd		Medium Type							
3rd		WP-Bit	0	0	0	0	0	0	EBC-Bit
4th		Block Descriptor Length							

Block Descriptor

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	0	0	0	0	0	0
2nd		Number of Blocks (MBS)							
3rd		Number of Blocks							
4th		Number of Blocks (LBS)							
5th		0	0	0	0	0	0	0	0
6th		Block Length (MSB)							
7th		Block Length							
8th		Block Length (LSB)							

Private Use Parameter(s)

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st									
nth									

The Sense Data Length field specifies the length in bytes of the following MODE SENSE data that is available to be transferred during the DATA IN phase.

Values for the Medium Type field shall be assigned as follows:

(00)	Default (only one medium type supported)
(01) to (7F)	Reserved for future standardization
(80) to (FF)	Private Use

If the first bit of the 3rd byte, the Enable Blank Check Bit (EBC) is set to ZERO blank checking of the medium during write operations is disabled. If the EBC Bit is set to ONE, blank checking during write operations is enabled. For Read-Only direct-access devices, the EBC Bit shall be set to ZERO.

If the 8th bit of the 3rd byte, the Write Protected Bit (WP) is set to ZERO, the medium is write enabled. If the WP Bit is set to ONE the medium is write-protected.

For Read-Only direct-access devices, the WP Bit shall be set to ZERO.

The Block Descriptor Length field specifies the length in bytes of all the Block Descriptors. It is equal to the number of Block Descriptors times eight and does not include the Private Use parameters, if any. If the length specified by the Block Descriptor Length field equals zero, no Block Descriptors shall be included in the parameter list. This condition shall not be considered as an error.

Each Block Descriptor specifies the medium characteristics for all or part of a logical unit. Each Block Descriptor contains a Number of Blocks field and a Block Length field.

The Number of Blocks field specifies the number of logical blocks that have the block length specified in the Block Descriptor Length field. The Block Length field specifies the length in bytes of each logical block.

6.7.2 Write-Once-Read-Multiple Optical Direct-access Devices

Name		Identifier
WRITE	Nm	(0A)
WRITE EXTENDED	B	(2A)
WRITE AND VERIFY	Nm	(2E)

B = mandatory in Basic Level

Nm = not mandatory in either level

6.7.2.1 WRITE Command (0A)

Specific command for optical direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	0	0	1	0	1	0
2nd	LUN			Logical Block Address (MSB)				
3rd	Logical Block Address							
4th	Logical Block Address (LSB)							
5th	Transfer Length							
6th	Private Use		0	0	0	0	Flag	Link

The WRITE command requests that the Target write the data transferred by the Initiator on the medium.

The Logical Block Address field specifies the address of the logical block where the write operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value specified by the Transfer Length field equals zero, 256 logical blocks shall be transferred. Any other value indicates the number of logical blocks to be transferred.

If an attempt is made to write a previously written block and blank checking is enabled, the command shall be terminated with a CHECK CONDITION status and, if the EXTENDED SENSE command is implemented, the Sense Key shall be set to BLANK CHECK.

6.7.2.2 WRITE EXTENDED Command (2A)

Specific command for optical direct access devices.

BYTE ↓ BIT→	8	7	6	5	4	3	2	1
1st	0	0	1	0	1	0	1	0
2nd	L U N			0	0	0	0	RA-Bit
3rd	Logical Block Address (MSB)							
4th	Logical Block Address							
5th	Logical Block Address							
6th	Logical Block Address (LSB)							
7th	0	0	0	0	0	0	0	0
8th	Transfer Length (MSB)							
9th	Transfer Length (LSB)							
10th	Private Use		0	0	0	0	Flag	Link

The WRITE EXTENDED command requests that the Target write the data transferred by the Initiator on the medium.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) is set to ONE, it indicates that the Logical Block Address field is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

The Logical Block Address field specifies the address of the logical block where the write operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of the data to be transferred. If the value of the Transfer Length field equals zero, no logical blocks shall be transferred. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be transferred.

If an attempt is made to write a previously written block and blank checking is enabled, the command shall be terminated with a CHECK CONDITION status and, if the EXTENDED SENSE command is implemented, the Sense Key shall be set to BLANK CHECK.

6.7.2.3 WRITE AND VERIFY Command (2E)

Specific command for optical direct access devices.

BYTE ↓	BIT→	8	7	6	5	4	3	2	1
1st		0	0	1	0	1	1	1	0
2nd		LUN			0	0	0	BC-Bit	RA-Bit
3rd		Logical Block Address (MSB)							
4th		Logical Block Address							
5th		Logical Block Address							
6th		Logical Block Address (LSB)							
7th		0	0	0	0	0	0	0	0
8th		Transfer Length (MSB)							
9th		Transfer Length (LSB)							
10th		Private Use		0	0	0	0	Flag	Link

The WRITE AND VERIFY command requests that the Target write the data transferred by the Initiator on the medium and then verify the data written.

When bit 1 of the 2nd byte, the Relative Address Bit (RA-Bit) field is set to ONE, it indicates that the Logical Block Address fiels is recorded as a two's complement displacement. This negative or positive displacement is to be added to the value of the Logical Block Address field last accessed on the Logical Unit to form the Logical Block Address. This feature is only available when linking commands. The feature requires that a previous command in the linked group has accessed a block of data on the Logical Unit.

When bit 2 of the 2nd byte, the Byte Check Bit (BC-Bit) is set to ZERO, the verification is simply a medium verification (CRC, ECC, etc.). When this Byte Check Bit is set to ONE, the verification is a byte-by-byte comparison of data written on the Peripheral Device and the data transferred from the Initiator.

The Logical Block Address field specifies the address of the logical block where the write operation shall begin.

The Transfer Length field specifies the number of contiguous logical blocks of data to be transferred. If the value of the Transfer Length field equals zero, no logical blocks shall be transferred. This shall not be considered an error condition. Any other value indicates the number of logical blocks to be transferred.

If an attempt is made to write a previously written block and blank checking is enabled, the command shall be terminated with a CHECK CONDITION status and, if the EXTENDED SENSE command is implemented, the Sense Key shall be set to BLANK CHECK.

7. COMMAND RESPONSES

7.1 Status

The Target sends Status as to the Initiator during the BUS STATUS phase; each Status consists of one byte of information. A Status is sent upon completion of each command, or after any abnormal conditions encountered during the command execution or if a Target cannot accept a command at this time. After the operation is cleared by an ABORT message, RESET BUS DEVICE message or HARD status will not be sent.

7.1.1 Status Byte

The Status byte is encoded as specified in the table below.

GOOD Status

The Target has successfully executed, an individual command or the last command in a chain of commands.

CHECK CONDITION Status

An abnormal or error condition has been encountered during the execution of a command. This Status could also be set by a device encountering the End of Media or a File Mark. The REQUEST SENSE command should be sent to determine the cause of this condition.

CONDITION MET Status

A SEARCH DATA command has been successfully completed, the block address may be obtained with the REQUEST SENSE command. This Status does not break a chain of linked commands.

BUSY Status

The Target is unable to accept a command from the Initiator at that time.

Note 5

The normal recovery action by the Initiator is to issue again the command at a later time.

INTERMEDIATE Status

A command within a sequence of linked commands has just been completed. If this Status is not sent the sequence of linked commands is broken.

RESERVATION CONFLICT Status

A command attempts to access a Logical Unit or an extent within a Logical Unit, for which that action has been reserved by another Initiator.

BIT	8	7	6	5	4	3	2	1	MEANING
	0	P	P	0	0	0	0	P	GOOD STATUS
	0	P	P	0	0	0	1	P	CHECK CONDITION
	0	P	P	0	0	1	0	P	CONDITION MET/GOOD STATUS
	0	P	P	0	0	1	1	P	Reserved for future standardization
	0	P	P	0	1	0	0	P	BUSY STATUS
	0	P	P	0	1	0	1	P	Reserved for future standardization
	0	P	P	0	1	1	0	P	Reserved for future standardization
	0	P	P	0	1	1	1	P	Reserved for future standardization
	0	P	P	1	0	0	0	P	INTERMEDIATE/GOOD STATUS
	0	P	P	1	0	0	1	P	Reserved for future standardization
	0	P	P	1	0	1	0	P	INTERMEDIATE/CONDITION MET/ GOOD STATUS
	0	P	P	1	0	1	1	P	Reserved for future standardization
	0	P	P	1	1	0	0	P	RESERVATION CONFLICT
	0	P	P	1	1	1	0	P	Reserved for future standardization
	0	P	P	1	1	1	1	P	Reserved for future standardization

Bits indicated by P are for Private Use, they are not specified by this Standard.

7.2 Sense Key

In response to a CHECK CONDITION Status a REQUEST SENSE command may be issued to obtain detailled error information. See 6.2.6.

APPENDIX A

GUIDELINES FOR ELECTRICAL CONNECTION

A.1 Selection between single-ended and differential option

The following factors should be considered when selecting between the two options:

	Single-ended	Differential
1.	6 m maximum cable length limit.	25 m maximum cable length limit.
2.	0,1 m maximum stub length.	0,2 m maximum stub length.
3.	All ports should be run from one power supply.	Each port can be run from its own power supply.
4.	May not be suitable for use in an electrically noisy environment.	Suitable for use in an electrically noisy environment.

All single-ended ports should be powered by a single supply because the device drivers and receivers when powered down singularly could be damaged by applied voltages, or could leak current which would corrupt data.

The single-ended option may not be suitable for use in an electrically noisy environment because of the lower characteristics impedance of screened cable.

A.2 Single-ended power distribution

All SCSI ports should be powered by a common supply. The connection between the power supply and each ports drivers and receivers should have a low impedance. For this reason it is recommended that multilayer PCB's are used with separate power and ground planes. There should be one local high frequency decoupling capacitor per driver chip, and one per receiver chip. All cable OV connections should have a low impedance connection to the PCB ground place.

A.3 Single-ended terminators

A resistor tolerance of ± 10% is recommended. The terminator supply voltage should be kept within a tight tolerance of 5,0 V to give improved noise immunity. The tolerance of the 5,0 V may be improved by decoupling the supply at the terminator; by ensuring that the length of the connection between the supply and the terminator is short if termination power is supplied via the cable; or by providing power to the cable via a separate low impedance connection.

A.4 Single-ended cables, drivers and receivers

For single-ended systems greater noise immunity can be achieved by using drivers with an output capability greater than the 48 mA specified. The effect of reducing the minimum output capability of the driver below 48 mA will be to limit the number of SCSI ports that may be attached or to reduce the distance over which the bus may be driven or reduce the transfer rate accross the bus due to increasing the cable skew delay. Similarly reducing the cable impedance below the 132 Ohm specified to a commercially available cable of approximately 100 Ohm will have the same effects. In order to operate the bus with 8 SCSI ports over 6 m with a cable of 100 Ohm impedance it is recommended that drivers with a minimum output capability of 60 mA be used. The Cable Skew delay will also be affected by the receiver input threshold. All receivers must have a minimum input hysteresis of 0,2 V. However, if this hysteresis is very large, problems could arise due to drivers struggling to reach very low or very high input threshold voltage levels. For this reason, schmitt-trigger receivers are not recommended.

A.5 Single-ended cable stubs

The maximum permissable stub length is defined to be 0,1 m. Stubs may cause problems by either reflecting glitches back towards the transmitting gate or by causing edges to be slowed down due to the capacitance of the stub. In order to minimize these problems it is recommended that the total capacitance of the cable stub, connector, driver and receiver should be less than 25 pF. When using a 6 m cable it is also recommended that the minimum stub spacing should be 0,4 m in order to prevent reflections from adjacent stubs superimposing to give large glitches. For short cables these recommendations may be relaxed as the bus will no longer exhibit transmission line behaviour.

A.6 Connecting and Disconnecting equipment from SCSI

Equipment may be plugged into or plugged from a live SCSI bus unless it receives terminator power via the cable. However, all single-ended ports should be run from a common power supply and equipment should not be connected to or disconnected from a live supply.

APPENDIX B

For applications requiring cables between cabinets or boxes it is recommended to use flat ribbon cable inside and if the cabinet or box is the EMI-screen then use a miniature ribbon connector (as used in the IEEE 488 - 1978 interface) when going through the screen.

The external cable may then be a round cable using the miniature ribbon connector with appropriate locking devices in both ends.

For boards in the EUROCARD boxes (IEC 297 or DIN 41494) the EMI-screen may be in the front of the boards and in this case a screened cable and connector (e.g. AMPHENOL FRC-50) may be attached through the front panel to the board connector specified in this ECMA Standard. It is thereby possible to use the flat ribbon cable specified in this ECMA Standard or the screened cable described above as required by the users.